AFMC PAMPHLET 63-103
15 JUNE 1994

Acquisition

Software Development Capability Evaluation

Volume 1

Department of the Air Force
Headquarters Air Force Materiel Command

BY ORDER OF THE COMMANDER
AIR FORCE MATERIEL COMMAND (AFMC)

AFMC PAMPHLET 63-103, Volume 1
15 JUNE 1994

Acquisition

SOFTWARE DEVELOPMENT CAPABILITY EVALUATION

Acquisition

SOFTWARE DEVELOPMENT CAPABILITY EVALUATION

This pamphlet provides guidance for planning and conducting a Software Development Capability Evaluation (SDCE). The SDCE is a structured methodology for assessing an organization's ability to develop software for mission critical computer resources. The primary purpose of the SDCE is to reduce acquisition risk for software-intensive systems. The SDCE is conducted as an integral part of the source selection process and addresses each offeror's ability to develop the software required by a specific request for proposal (RFP). The evaluation covers the total software development process, including systems and software engineering, management, quality, product control, organizational support, tools, facilities, and personnel experience and qualifications. Risk reduction is achieved by increasing the probability of selecting a fully capable offeror with the capacity to develop software consistent with the RFP requirements and program baselines, by early and comprehensive visibility into the offeror's proposed capabilities, and by ensuring contractual commitment by the offeror to use the processes proposed. The SDCE is intended to be applied to both prime contractors and their associates and to subcontractors responsible for software development. This publication does not apply to the U.S. National Guard or U.S. Air Force Reserve units and members.

This pamphlet is published in two volumes. Volume 1 provides a detailed description of the SDCE methodology, including a comprehensive step-by-step expanation of how to perform the SDCE, and a detailed description of the model, including criteria and questions. Volume 2 contains an assortment of support material such as examples, templates, forms, checklists, and briefing charts that facilitate the use of the SDCE method.

OPR: HQ AFMC/ENPC (Mr William J. Urschel)
Certified by: HQ AFMC/ENP(Col Louis E. Mitchell)

Pages: 271/Distribution: F;X
AUL/LDEA, Maxwell AFB AL 36112-5564 - 1

Figures

Tables

CHAPTER 1. INTRODUCTION

The Software Development Capability Evaluation is a method to be applied as an integral part of the source selection process for systems and subsystems which include mission critical computer resources (MCCR). The SDCE produces a review of each offeror's software development process within the context of the systems engineering and management process. It focuses on the offeror's specific capabilities and capacities to successfully develop the software required on the program entering source selection. This introduction covers background, purpose, policy, role in source selection, scope and applicability, benefits and limitations, and ownership and support.

1.1 Background

Software acquisition and development have become major challenges in the successful development, operation, and support of Air Force Materiel Command (AFMC) developed systems. Software technology not only implements the functionality associated with electronic-based systems and subsystems, it has become the key technology in integrating the system functions. Software development is integral to all mission critical systems developed within AFMC, whether airborne, ground based or space based.

While mission critical software has grown dramatically in complexity and magnitude, the systems and software engineering management discipline necessary to successfully develop software has not kept pace. AFMC product and logistic centers, together with defense contractors, are faced with the challenge of developing large and complex software systems that are critical to system performance, without well-defined and consistently applied systems engineering, software engineering, and management discipline. This lack of engineering discipline, and the lack of an adequate evaluation process, have adversely impacted numerous AFMC programs. To meet this challenge, defense contractors have independently developed internal standard engineering practices, procedures, methods, and tools. However, these have been applied with varying degrees of consistency and success. Even when a software development practice or procedure is defined in a company-approved document, it is not always implemented on that company's programs.

This situation has created a problem in the acquisition process of selecting a fully qualified contractor, or contractor team, for programs that involve significant software development. The specific problem is two fold: (1) how to determine the technical ability of an offeror to successfully develop the software to program requirements within program baselines, thereby reducing acquisition risk, and (2) how to determine whether the offeror has technical and management processes in place that will insure consistent execution at the highest possible capability.

The SDCE contributes to solving this problem by establishing a methodology to evaluate an offeror's software engineering and management capability in the context of the source selection process. The SDCE also evaluates the offeror's systems engineering capability, which directly impacts software, by including systems requirements definition and allocation to software and the multiple levels of integration and testing required to complete the systems and software development. Major characteristics of the SDCE are summarized in table 1-1.

Table 1-1. SDCE Major Characteristics

- Complete but flexible coverage of management and technical issues.
  - Structured model is readily tailorable for specific program.
  - Strong focus on the program at hand.
- Well integrated into the source selection process.
  - Systematic SDCE activity process flow.
  - SDCE team members part of SSEB.
  - Accommodates acquisitions with "no discussions."
  - Determines strengths, weaknesses, and risks.
- Bidders describe and commit to their process capabilities.
  - Integrated with general proposal material (technical and management volumes).
  - Specific SDCE material submitted with proposal.
  - Commitment to process in SDP, SEMP, and SEMS.
- Comprehensive data-gathering set of questions and criteria.
- Bidders provide evidence of process capability application.
- Rationale and justification for proposed new capabilities.
- Site visit dialogue for understanding bidder's capability.
- Promotion of industry and government cooperation.
  - Complete SDCE documentation including guidelines, forms, and templates.
  - Mechanisms to ensure consistent application.
  - Public methodology and criteria.
- Tailorable to be efficient and effective.

This SDCE method was developed by an AFMC and Industry Process Action Team (PAT) chartered by AFMC/EN. The method was derived primarily from the ASC/EN Software Development Capability/Capacity Review (SDCCR), and from the Software Engineering Institute’s (SEI) Software Capability Evaluation (SCE), which is based on the SEI Capability Maturity Model (CMM). In addition, the PAT added coverage for areas critical to successful software development not found in the existing evaluation methodologies. The process for applying the SDCE within the source selection environment also contains many new features and attributes.

1.2 Purpose of the SDCE

The primary purpose of the SDCE is to increase the probability of selecting an offeror (team) capable of successfully developing software to meet specified RFP requirements within program baselines. The SDCE is used to evaluate the offeror's specific capability to develop the software on a particular program within the context of the program management and systems engineering process. A second purpose of the SDCE is to elicit a contractual commitment to implement methods, tools, practices, and procedures which form and support a necessary software engineering discipline. The major purposes of the SDCE are summarized in table 1-2.

Table 1-2. SDCE Major Purposes

- Support the source selection process.
- Reduce program execution risk.
- Identify strengths that contribute to successful program development.
- Identify weaknesses and risks (to resolve early with the winning contractor).
- Provide enhanced focus and concentration on total software and related systems engineering capability.
  - Processes
  - Personnel
  - Tools and techniques
  - Facilities and infrastructure
- Understand the bidder's ability and commitment to perform as proposed.
  - Institutionalization or past experience
  - Contractual commitment
- Ensure good software and related systems engineering processes are followed.

1.3 Policy

Air Force policy on MCCR requires a "preaward survey" of the offeror's resources and capabilities to perform and manage software development. This is defined in AFR 800-14, Attachment 6. In addition, AFSCP 800-51, Software Development Capability Assessment, discusses the SDCCR and SCE methods which this SDCE method supersedes as the single AFMC approach to satisfying the policy stated above.

1.4 Role in Source Selection

The SDCE plays an essential and integral role in the source selection structure of Areas, Factors, and Subfactors; the SDCE is typically a separate Factor under the Technical Area. The SDCE team bases its evaluation on information submitted with the proposal, and upon information gathered during site (plant) visits conducted as part of the source selection. The information is assessed against pre-defined source selection criteria and standards; it is evaluated in context by the Source Selection Evalation Board (SSEB); and it is reported in a manner consistent with the prescribed source selection evaluation procedures.

The SDCE method is organized into six Functional Areas: Program Management, Systems Engineering, Software Engineering, Quality Management and Product Control, Organizational Resources and Program Support, and Program Specific Technologies. These Functional Areas collectively cover the essential development capabilities and processes necessary to successfully develop software in the context of a system development. The SDCE supports establishing a commitment to a planned event driven engineering development process consistent with the Software Development Plan (SDP), Systems Engineering Master Schedule (SEMS), and Systems Engineering Management Plan (SEMP).

1.5 Scope and Applicability

The SDCE is intended to be applied to all programs acquired under DoDD 5000.1 and DoDI 5000.2, where software development is important to successful system development. The SDCE method covers all newly developed software, modified software, incorporation of commercial off-the-shelf (COTS) and reused software, and the integration of all software into a functioning system. The SDCE method evaluates software engineering and development processes along with the systems engineering and development disciplines that are directly involved in MCCR software development. Systems engineering is emphasized because software engineering is an integral part of the systems engineering process for MCCR applications.

The SDCE is primarily applicable to source selections for the Engineering and Manufacturing Development (EMD) phase. However, it is also applicable to major modification programs and to Demonstration/Validation (Dem/Val) phase source selections when the software is likely to be used in a follow-on contract phase. The SDCE is intended to be applied with prime contractors and with associate contractors and subcontractors who are planning to develop significant software integral to the system. SDCEs should be used for sole source as well as competitive contracts. Prime contractors may use the technical material (minus the government source selection-specific elements) to evaluate subcontractor capabilities when subcontractors are selected by the primes, after prime contract award.

1.6 Benefits and Limitations

Although the SDCE method assists the Source Selection Evaluation Board in selecting an offeror capable of the software development and elicits a contractual commitment from the offeror to apply its capabilities (methods, tools, practices, etc.), the SDCE has certain limitations in its role during source selection. This subsection outlines the benefits of the SDCE method as well as these limitations.

Benefits of the SDCE Method

- Requires a comprehensive description of the software development capabilities in terms of engineering and management processes, methods, tools, and resources.
- Reviews the systems engineering and other development disciplines and processes directly related to software development.
- Reviews and supports gaining a commitment to follow well defined and planned processes described in the software development plan and tied directly to the systems engineering master schedule.
- Provides a vehicle for a comprehensive dialogue between the contractor’s proposal team and the acquisition program source selection team, addressing the processes, methods, practices, and tools to be applied in executing the program development. A side benefit of this is the team building and mutual understanding of the offeror’s capability and process that is facilitated through the site visit.

- Emphasizes with the offeror the importance to the acquisition program office of having capabilities and processes in place to successfully develop software.
- Reduces program risk through early focus (during source selection) on software capability and process. The SDCE is a problem prevention technique as compared with a cure technique.

Limitations of the SDCE Method

- Does not establish software program requirements (statement of work, specifications, Contract Data Requirements Lists).
- Will not assure realistic, achievable software development schedules will be established and followed. The SDCE will, however, review the offeror's estimating methods and basis for the proposed software development schedules.
- Does not address the specific software design solution to the program requirements. This design is described in the technical proposal and should be reviewed by the technical evaluation panel.
- Cannot assure that the offeror's resources, in terms of personnel, staffing, and facilities, will in fact be applied on the program after contract award.

1.7 Ownership and Support

An office of primary responsibility (OPR) for the SDCE method has been established at Headquarters Air Force Materiel Command. SDCE focal points have also been established at each Logistics and Product Center. These local center offices are available to assist program offices and SDCE teams in tailoring and applying the SDCE method. They are also responsible for improving the method as experience dictates.

In summary, it is essential to use the SDCE as one of several, interrelated methods to achieve software development success. The SDCE must be used with related initiatives such as systematic software development metrics; software development plans that define and describe the processes, methods, practices, and tools to be applied on the program; and the systems engineering management plan.

CHAPTER 2. OVERVIEW OF THE SDCE METHOD

The SDCE method has two constituent parts, a model that structures the SDCE criteria and questions, and a process for applying the model to a given source selection. Subsection 2.1 summarizes the SDCE model and provides background information about its use and legacy. Subsection 2.2 summarizes the process for applying SDCE within the source selection environment. Chapter 3 provides more detail on the model, and chapter 4 provides complete descriptions of activities and tasks for the SDCE process. The model criteria and questions are tabulated in chapter 5.

Many specific terms and phrases (both related to the SDCE and to source selection in general) are used throughout this pamphlet. Comprehensive definitions can be found in the glossary, attachment 1.

Source selection is a complex environment with many different processes, products, and players. Figure 2-1 shows at a high level how the SDCE method interacts with and relates to the major source selection objects.

2.1 SDCE Model Overview

2.1.1 Background

The SDCE model focuses on the critical capabilities that historically have represented areas of high risk and that may be critical when software-intensive systems are being developed. An appropriate subset of the model should be used in evaluating the adequacy of capabilities proposed for a specific program source selection. The model is an organized set of software and software-related system development capabilities that acquisition organizations use as a basis for evaluating an offeror's software development capability and capacity. For a specific source selection, the model facilitates the identification of an offeror's strengths, weaknesses, and risks. The model is intended to be non-prescriptive for the process required to achieve a critical capability.

The SDCE model is primarily based on the Capability Maturity Model, developed by SEI, and the Software Development Capability/Capacity Review, developed by ASC/EN(CR). These models have been used in many Air Force acquisitions and process improvement efforts and have been subjected to extensive community reviews. They provide a basis for the state of the practice in software development and lessons learned on the criticality of specific practices. The SDCE questions were also drawn primarily from the SDCCR and CMM.

The SDCE model, however, corrects some of the shortfalls of the CMM and SDCCR. For instance, the CMM does not address systems engineering interfaces and human resources and is focused on organizational versus program-specific capabilities, and the SDCCR does not address process improvement efforts, defect prevention, metrics, and technology assessment and transition. The SDCE model is consistent with the process improvement efforts based on these models, addresses systems engineering and human resources, and focuses on program specifics.

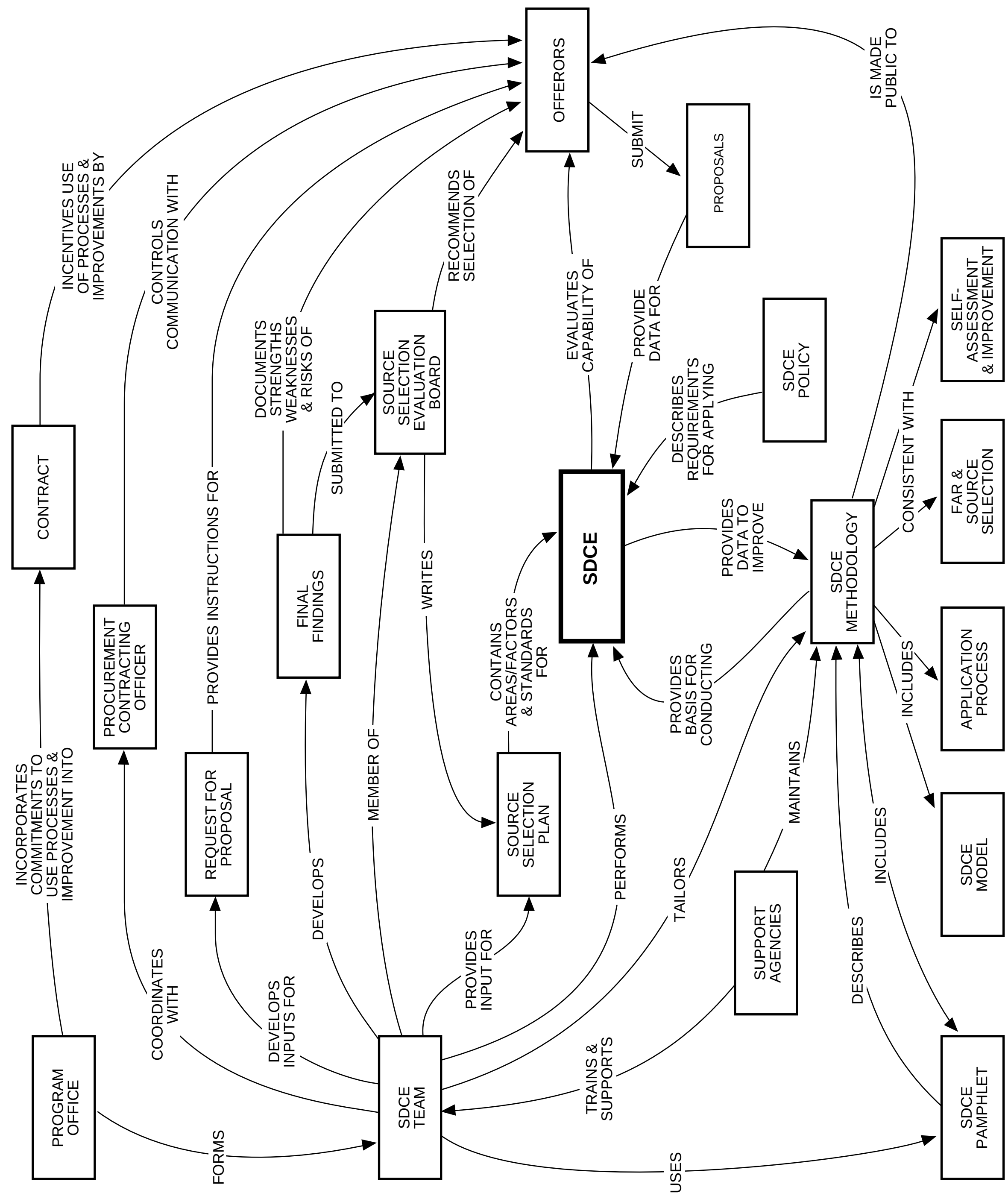


Figure 2-1. The SDCE Method in Context

2.1.2 Structure

The model consists of a structure, a set of model criteria, and questions (figure 2-2). The structure is a hierarchical decomposition (Functional Area, Critical Capability Area, Critical Capability) of the capabilities that have been identified as potential discriminators in the source selection process for software-intensive systems. The structure facilitates the rollup of identified strengths, weaknesses, and risks and the tailoring of criteria for a particular source selection. The model criteria are the bases for evaluating the adequacy of a specific aspect of an offeror's capability. Strengths, weaknesses, and risks are initially identified against the model criteria and then rolled up to the level defined by the source selection standards. The questions are designed to give insight into an offeror's processes, methods, and tools. The questions are traceable to criteria to assist the SDCE evaluation team in its tailoring analysis. The model criteria and associated questions are structurally contained within the Critical Capabilities (CCs), as shown in figure 2-3.

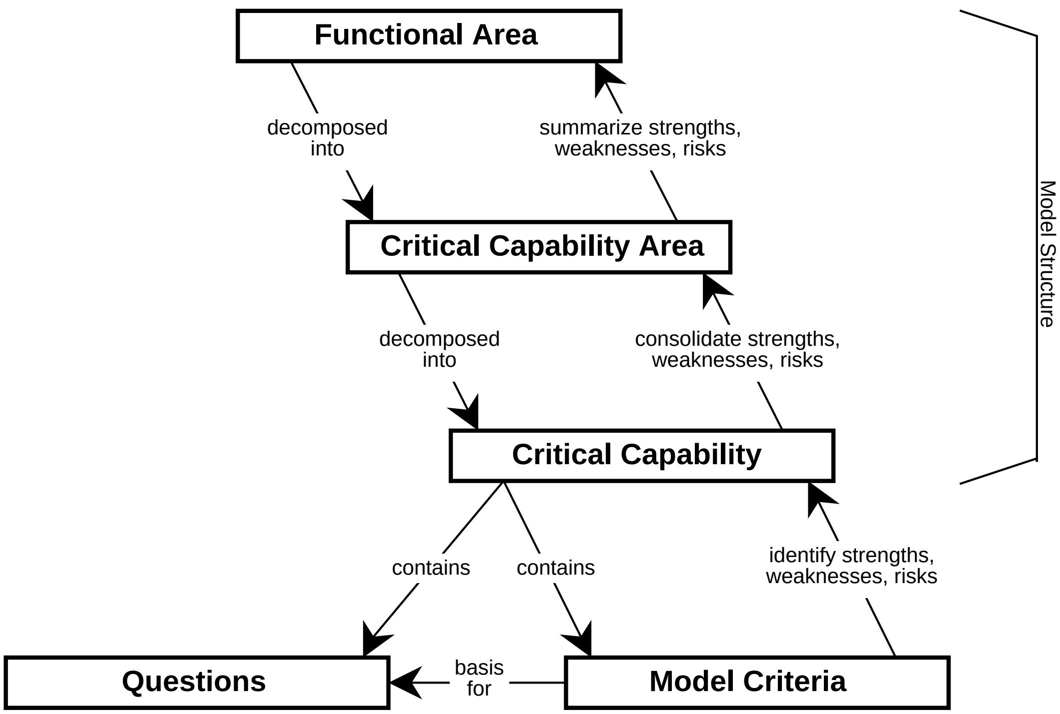


Figure 2-2. The SDCE Model

The model criteria represents a standard and repeatable way to evaluate an offeror's capabilities in a source selection. The criteria identify strengths and weaknesses within the context of the source selection evaluation standards.

The questions are designed to elicit the information necessary to provide insight into the offeror's capability and capacity. This information is used in evaluating the offeror against the standards for evaluation using the model criteria as guidance. The questions are derived from the model's criteria and also support "no discussion" decisions. The questions provide a standard approach to gathering data and should provide enough information to evaluate the offeror under "no discussion" constraints.

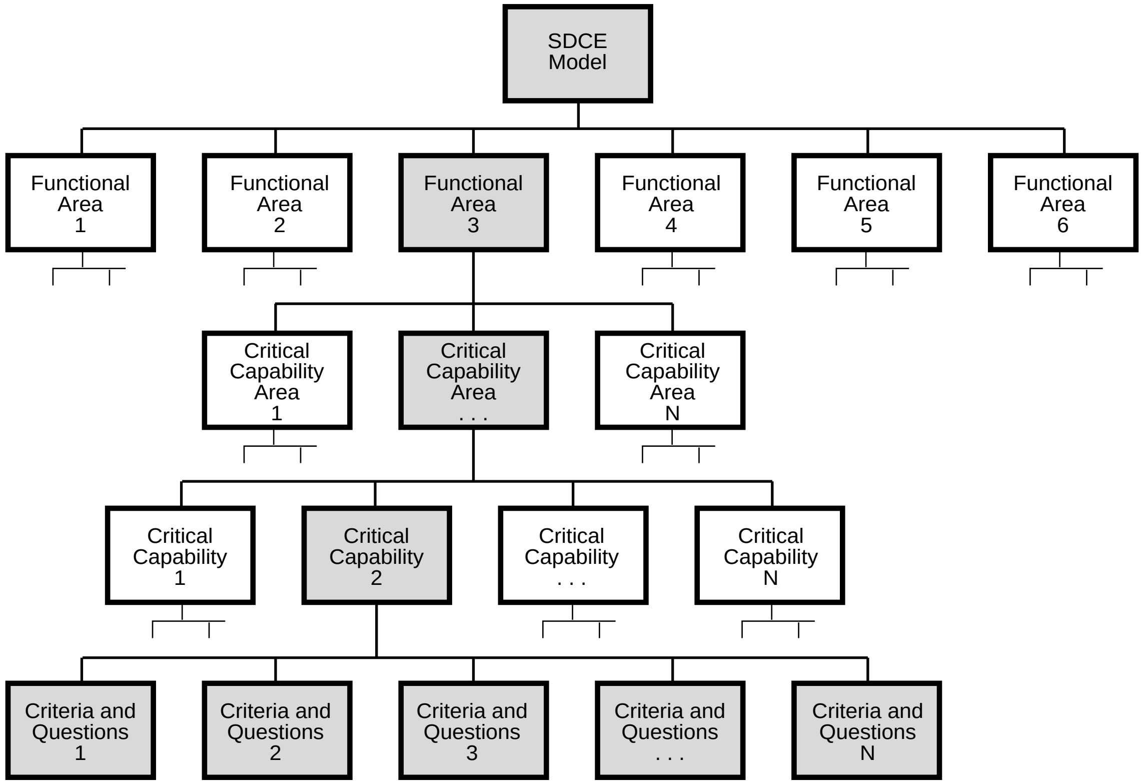


Figure 2-3. Generic SDCE Model Structure

2.1.3 Functional Area Summaries

The model consists of six interrelated Functional Areas (FAs) shown in figure 2-4. Three of the functional areas are layered and nestled, with "Software Engineering" being part of "Systems Engineering" which is, in turn, part of "Program Management." The other three Functional Areas cut across the first three FAs to various degrees. For example, because training applies to all of the first three FAs, it was placed in FA 5 rather than repeat the training coverage. Given these interrelationships, Critical Capability Areas (CCAs) have been placed within the six FAs based on a "best fit" determination. That is, some CCAs and Critical Capabilities are not clear, conceptual subsets of a single FA and may seem to cut across multiple FAs. For the purposes of this model, however, each CCA has been assigned to a single FA.

All the model's Functional and Critical Capability Areas are shown in figure 2-5. The FAs are summarized in the following paragraphs.

Functional Area 1, Program Management. The purpose of this Functional Area is to evaluate program level management capabilities that relate closely to successful software development and management. Embedded software development is highly dependent upon and integrated with the system level development capabilities. It is essential that these program level management processes and procedures are established, and that they are consistent and compatible with the software engineering processes and procedures. Program Management should provide visibility into the

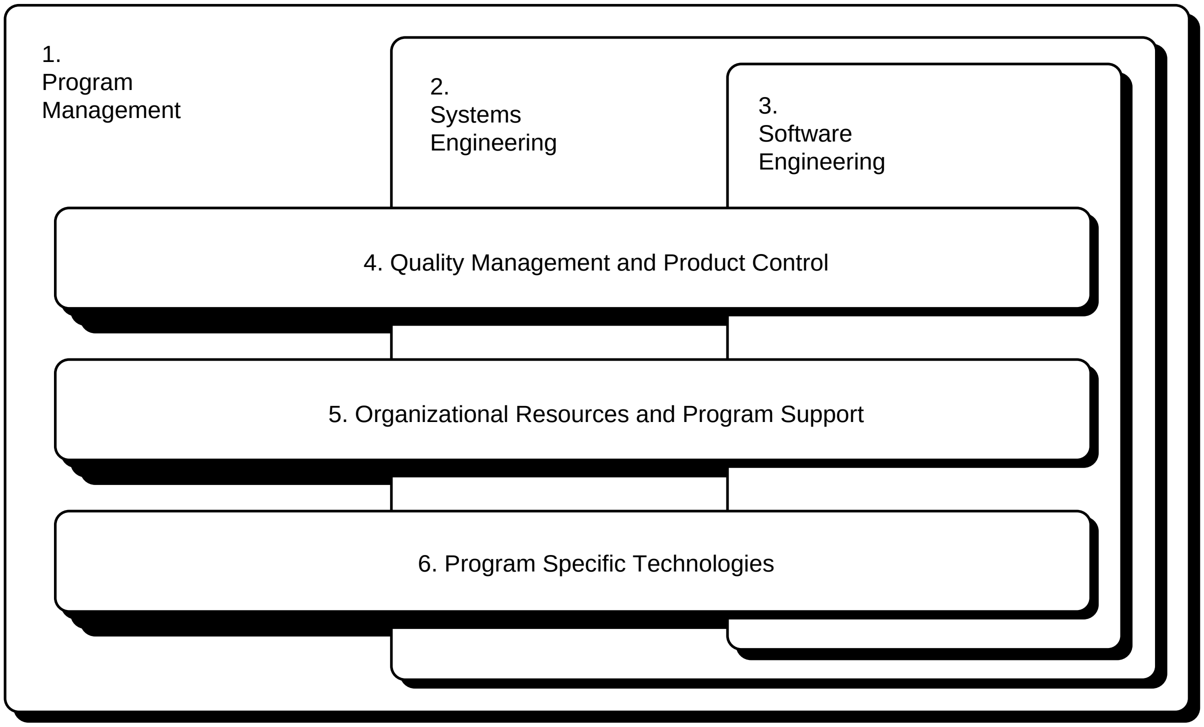


Figure 2-4. SDCE Functional Areas

actual cost, schedule, and technical progress of the program. This Functional Area encompasses the following Critical Capability Areas:

- 1.1 Management Authority, Responsibility, and Accountability
- 1.2 Program Planning and Tracking
- 1.3 Subcontractor Management
- 1.4 Legal and Contracting Issues
- 1.5 Risk Control

Functional Area 2, Systems Engineering. The purpose of this Functional Area is to focus attention on those aspects of systems engineering that have the greatest potential impact on a successful software development effort in terms of development schedule, development and life cycle cost, system quality, and support for a fielded system that meets user needs. This Functional Area encompasses the following Critical Capability Areas:

- 2.1 System Requirements Development, Management, and Control
- 2.2 Computer System Architecture Design and Review Process
- 2.3 Supportability
- 2.4 Intergroup Coordination
- 2.5 Systems Engineering Planning
- 2.6 System Integration and Test
- 2.7 Reuse

Functional Area 3, Software Engineering. The purpose of this Functional Area is to evaluate capabilities for the management and engineering development of the software product. This FA

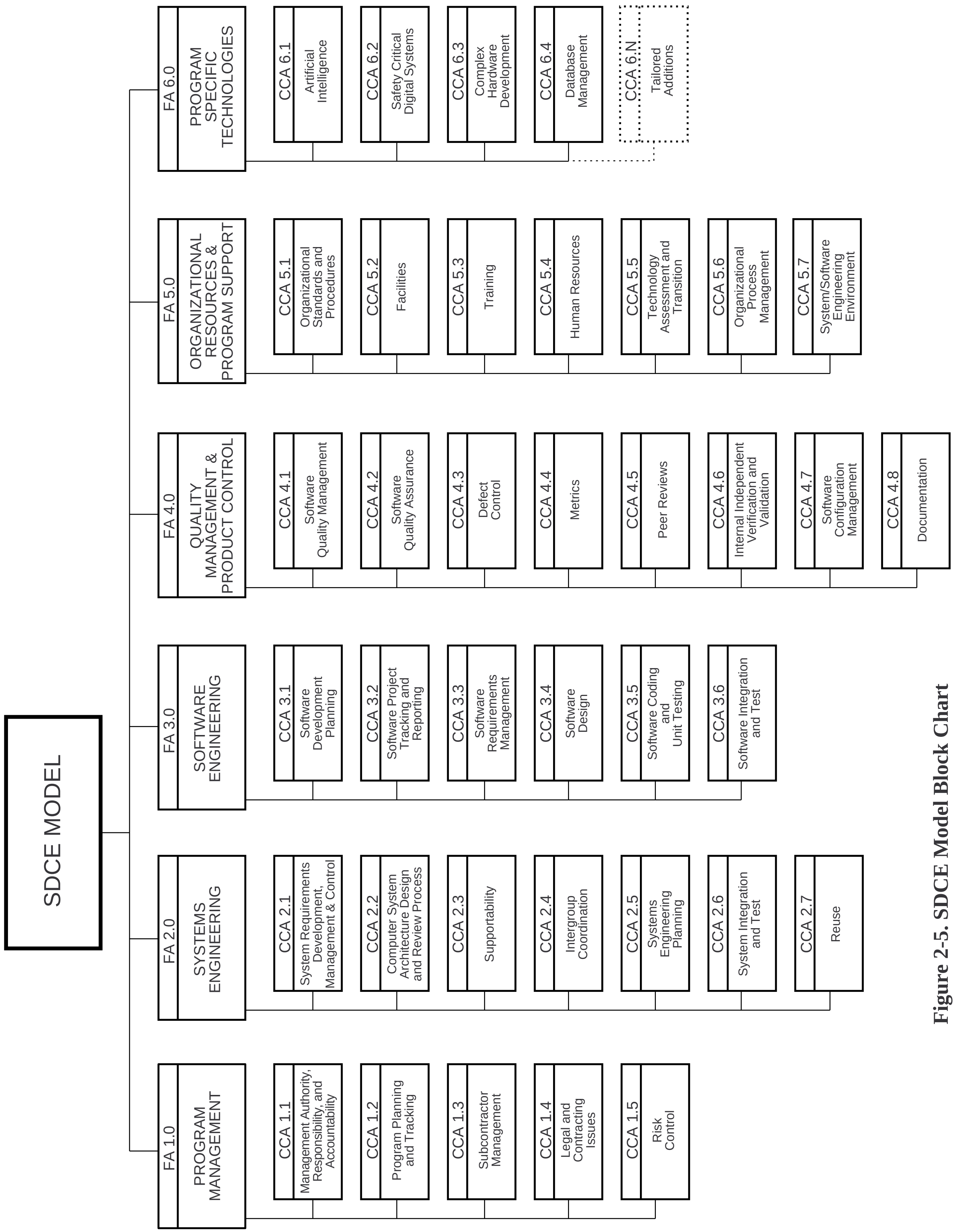


Figure 2-5. SDCE Model Block Chart

focuses attention on generation of the software development plan; estimation of size, cost, and schedule; definition of development methodologies; tracking and reporting against the plan; and development and control of software requirements, design, code, integration, and testing. This Functional Area encompasses the following Critical Capability Areas:

3.1 Software Development Planning
3.2 Software Project Tracking and Reporting
3.3 Software Requirements Management
3.4 Software Design
3.5 Software Coding and Unit Testing
3.6 Software Integration and Test

Functional Area 4, Quality Management and Product Control. The purpose of this Functional Area is to assure the quality of the program's software products and establish and maintain their integrity throughout the program's life cycle. Quality Management involves defining, planning, implementing, and monitoring quality goals. Product Control involves identifying the software configuration, systematically controlling changes to the configuration, developing documentation, and maintaining the integrity and traceability of the configuration throughout the life cycle. This Functional Area encompasses the following Critical Capability Areas:

4.1 Software Quality Management
4.2 Software Quality Assurance
4.3 Defect Control
4.4 Metrics
4.5 Peer Reviews
4.6 Internal Independent Verification and Validation (IIV&V)
4.7 Software Configuration Management
4.8 Documentation

Functional Area 5, Organizational Resources and Program Support. The purpose of this Functional Area is to evaluate organizational resources to the extent they are applied to support the program at hand. When evaluating the organizational resources against the criteria, the emphasis should be on those specific organizational resources which will be applied to the program, not on all the organizational resources. This Functional Area encompasses the following Critical Capability Areas:

5.1 Organizational Standards and Procedures
5.2 Facilities
5.3 Training
5.4 Human Resources
5.5 Technology Assessment and Transition
5.6 Organizational Process Management
5.7 System/Software Engineering Environment

Functional Area 6, Program Specific Technologies. The purpose of this Functional Area is to address technologies or application areas which are not required on a wide range of program developments. These CCAs need to be tailored out if no application to the program is envisioned.

Conversely, additional CCAs for unique technology or application areas applicable to the program may need to be developed. This Function Area encompasses the following four Critical Capability Areas and can be expanded as necessary to meet program needs:

- 6.1 Artificial Intelligence
- 6.2 Safety Critical Digital Systems
- 6.3 Complex Hardware Development
- 6.4 Database Management

2.2 SDCE Process Overview

2.2.1 Introduction

In addition to providing a comprehensive structure of capabilities, criteria, and questions, the SDCE method includes a detailed process for applying the model and evaluating contractor capability, commitment, and experience for a particular source selection. This process, from the earliest stages of conception through contract award and subsequent follow-up, is documented in this pamphlet. A flow diagram, showing the thirteen top-level activities and their interrelationships, is presented in figure 2-6. To help distinguish them from the six Functional Areas of the SDCE model, these activities are identified with letters, A through M.

Each of the thirteen top-level activities is further broken down into multiple constituent tasks or items in Figure 2-7. Detailed instructions and guidelines for the entire process are covered in chapter 4. The rest of this subsection provides a summary of the activities and a list of key tasks for each.

2.2.2 Activity Summaries

Activity A, Determine Applicability. The application of the SDCE must be considered as early as possible in the acquisition cycle. This usually begins with a local advocacy office (for example, the Center SDCE OPR) or support group and is augmented with program office personnel and specifically selected team members as the process unfolds. Key tasks for this activity are:

- A.1 Begin Acquisition
- A.2 Develop Initial Awareness
- A.3 Familiarize Responsible Program Office Personnel with SDCE
- A.4 Determine Applicability of SDCE to Acquisition
- A.5 Promote SDCE and Obtain Commitment to Use

Activity B, Select and Prepare Team. The SDCE team must be selected carefully. For efficiency, a small streamlined team is desired. Technical depth, coverage, and experience are essential for effectiveness. These potentially conflicting requirements must be adequately balanced. Training of the SDCE team in both the content of the overall source selection and the SDCE method is essential. Additionally, the planning and tailoring tasks may highlight a need to acquire and train additional team members. Key tasks for this activity are:

- B.1 Select Team Leader
- B.2 Define Team Size and Makeup

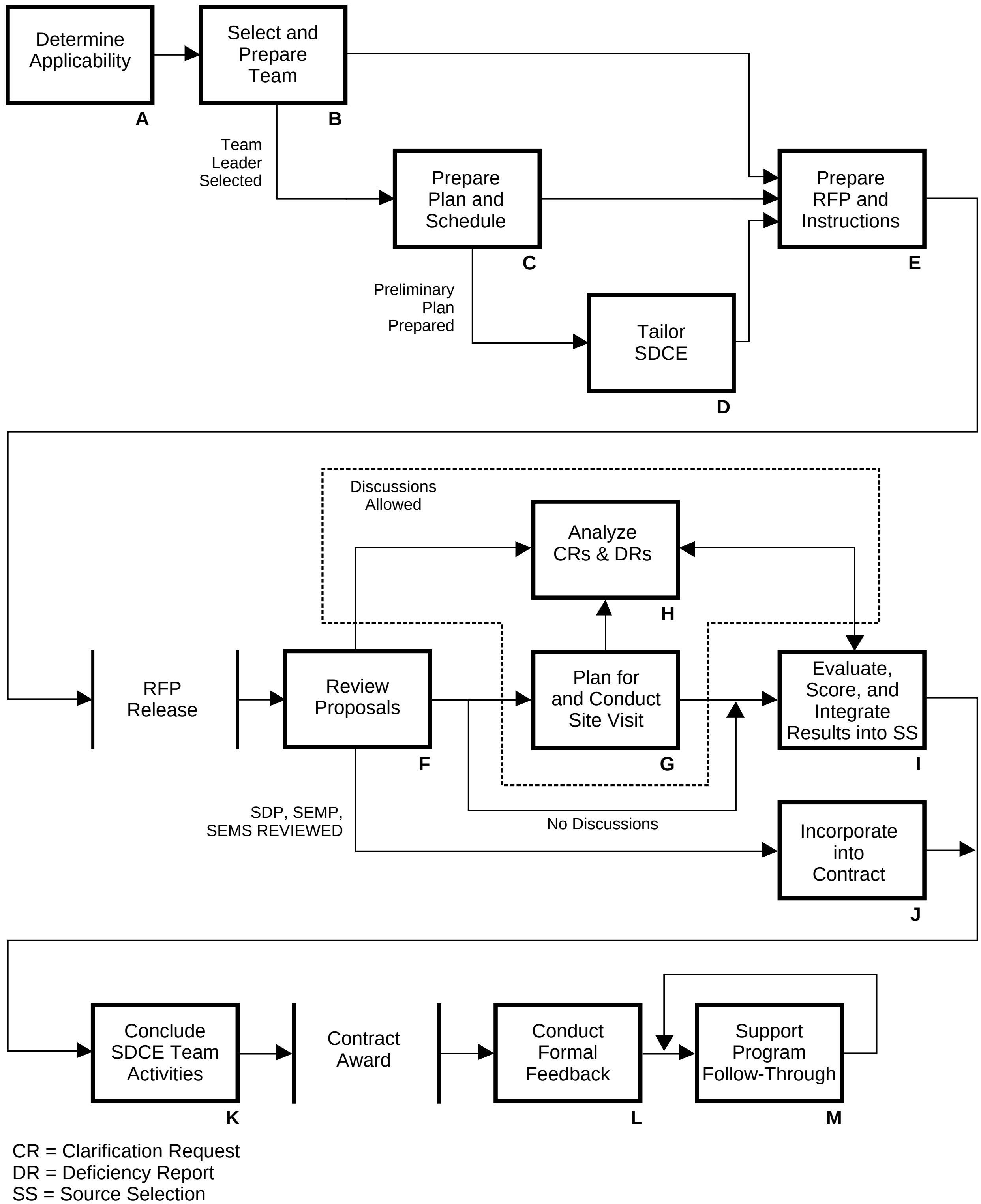


Figure 2-6. SDCE Activity Flow

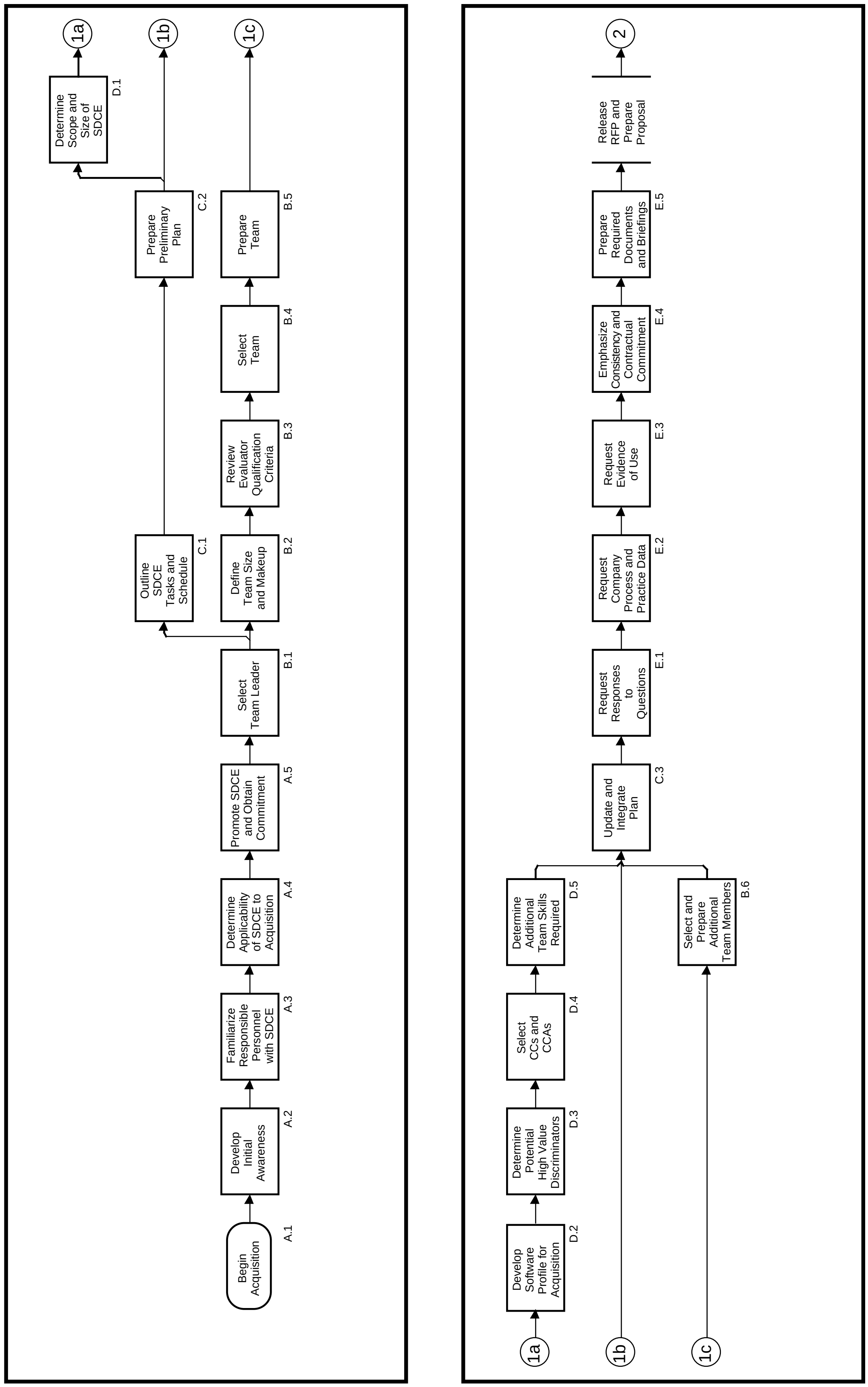


Figure 2-7. SDCE Task Flow

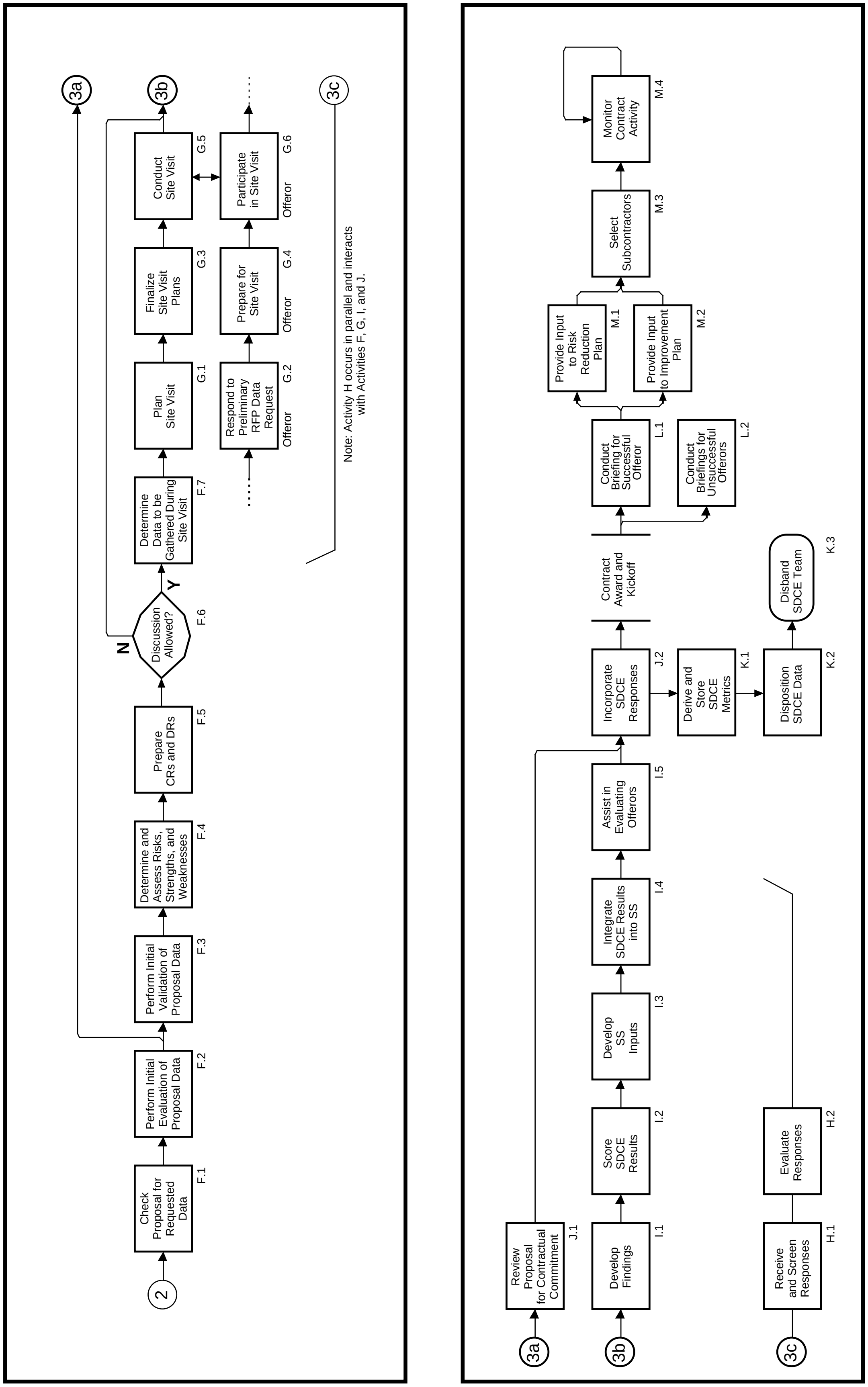


Figure 2-7. SDCE Task Flow (Continued)

B.3 Review Evaluator Qualification Criteria
B.4 Select Team
B.5 Prepare Team
B.6 Select and Prepare Additional Team Members

Activity C, Prepare Plan and Schedule. As soon as the initial SDCE team members have been identified, specific planning for the SDCE should begin. When the complete team is in place, the planning can be finalized. Of particular importance are the related tasks of determining the relationship of the SDCE model with the overall source selection structure and the preparation of specific evaluation standards for the source selection at hand. Although most of this planning is SDCE-specific, careful coordination with the overall source selection planning is required. Consideration should be given to integrating the SDCE planning information into the overall source selection planning, when possible, rather than using stand-alone SDCE documents. Key tasks for this activity are:
C.1 Outline SDCE Tasks and Schedule
C.2 Prepare Preliminary Plan
C.3 Update and Integrate Plan

Activity D, Tailor SDCE, Select Criteria and Questions. The SDCE model is large and comprehensive and must be tailored to focus on the high value discriminators for the program at hand. Tailoring not only includes the selecting of CCAs, CCs, criteria, and questions, but may also include the addition of specific program-unique capabilities not covered in the current version of the model. Mechanisms are provided for coordination of proposed updates with the AFMC SDCE OPR to facilitate SDCE process improvements. Key tasks for this activity are:
D.1 Determine Scope and Size of SDCE
D.2 Develop Software Profile for this Acquisition
D.3 Determine Potential High Value Discriminators
D.4 Select CCs and CCAs
D.5 Determine Additional SDCE Team Skills Required

Activity E, Prepare RFP and Instructions. The SDCE method requires the RFP to contain very specific instructions to the offeror. Additional information may be required in the statement of work (SOW) and Contract Data Requirements List (CDRL). The SDCE team must work closely with the larger source selection team in preparing these key documents. Key tasks for this activity are:
E.1 Request Responses to Questions
E.2 Request Company Process and Practice Data
E.3 Request Evidence of Use
E.4 Emphasize Consistency and Contractual Commitment
E.5 Prepare Required Documents and Briefings

Activity F, Review Proposals. Initial proposal review can begin as soon as the proposals are received. The primary purpose of this activity is to perform enough analysis to support a "competitive range" decision. Part of the decision will be whether or not discussions with the offerors will be allowed. Key tasks for this activity are:
F.1 Check Proposal for Requested Data
F.2 Perform Initial Evaluation of Proposal Data

F.3 Perform Initial Validation of Proposal Data
F.4 Perform Initial Assessment of Strengths, Weaknesses, and Risks
F.5 Prepare CRs and DRs
F.6 Release CRs and DRs if Discussions Allowed
F.7 Determine Data to be Gathered via SDCE Site Visit

Activity G, Plan for and Conduct Site Visit. A site visit is, by regulation, a discussion. Therefore, sites are visited only if it is determined that discussions are necessary for this particular procurement. Due to the intensive give-and-take nature of this activity, the offeror tasks have been interleaved with the government tasks in the SDCE process flow and descriptions. These government (evaluation team) and offeror tasks are:

G.1 Evaluation Team: Plan Site Visit
G.2 Offeror Team: Respond to Preliminary RFP Data Request
G.3 Evaluation Team: Finalize Site Visit Plans
G.4 Offeror Team: Prepare for Site Visit
G.5 Evaluation Team: Conduct Site Visit
G.6 Offeror Team: Participate in Site Visit

Activity H, Analyze Clarification Requests and Deficiency Reports. Throughout the evaluation process (primarily in activities F and I), CRs and DRs may be released to the offerors. When the responses are received, they must be processed, analyzed, and dispositioned. Follow-up CRs or DRs may be necessary. Key tasks for this activity are:

H.1 Receive and Screen Responses
H.2 Evaluate Responses

Activity I, Evaluate, Score, and Integrate Results into Source Selection. The SDCE process provides a structured method for the analysis and determination of strengths, weaknesses, and risks at the CC level. The analysis is based on an evaluation of the offeror's ability to meet the SDCE model criteria as well as other considerations such as commitment to use the proposed approach and evidence of past application of the approach. The detailed findings are then rolled up into the source selection evaluation structure. As an integral part of this activity, the findings are compared to the evaluation standards, color codes and risk ratings are assigned, narratives are written, and results are coordinated with other source selection evaluation teams. Key tasks for this activity are:

I.1 Develop Findings
I.2 Score SDCE Results
I.3 Develop Source Selection Inputs
I.4 Integrate SDCE Results into Source Selection
I.5 Assist in Evaluating Offerors

Activity J, Incorporate into Contract. One of the fundamental features of the SDCE method is the development of a contractual commitment to the capabilities proposed. This activity focuses on ensuring that the various contractual documents are in place, are adequate, and have been updated with the results of responses to CRs and DRs that were processed during the source selection period. Key tasks for this activity are:

J.1 Review Proposal for Contractual Commitment
J.2 Incorporate SDCE Responses

Activity K, Conclude SDCE Team Activities. At this point, the major activities of the SDCE team are complete. With the exception of the program office personnel who will transition to the execution phase of the contract, and possibly the SDCE team leader, the team can be disbanded. Key tasks for this activity are:

K.1 Derive and Store SDCE Metrics
K.2 Disposition SDCE Data
K.3 Disband SDCE Team

Activity L, Conduct Formal Feedback. Consistent with the source selection regulations, the results of the SDCE activities must be available for feedback to the various bidders. Coordination with the local contracting office is essential. The SDCE team leader and/or the program office personnel who were part of the SDCE team will participate in the preparation and conduct of the briefings. Key tasks for this activity are:

L.1 Conduct Formal Feedback Briefing for Successful Offeror
L.2 Conduct Formal Feedback Briefings for Unsuccessful Offerors

Activity M, Support Program Follow-Through. The SDCE method is focused on the source selection, but results developed during the source selection can become the basis for program follow-on tasks. Key areas where the SDCE can support long-term contract execution are:

M.1 Provide Input to Risk Reduction Plan
M.2 Provide Input to Improvement Plan
M.3 Select Subcontractors
M.4 Monitor Contract Activity

CHAPTER 3. DESCRIPTION OF THE SDCE MODEL

In the SDCE method, capabilities needed for large-scale embedded software acquisitions are structured into a three-level hierarchy or model. The lowest level of the hierarchy contains Critical Capabilities. Associated with each of these Critical Capabilities is a set of criteria describing measures of goodness for that capability and a set of questions for determining a contractor's approach with respect to that capability. Related Critical Capabilities are grouped into a Critical Capability Area; related Critical Capability Areas are then grouped into a Functional Area. This chapter, organized by the model's Functional Areas, describes the purpose and function of each of the Critical Capability Areas. The Critical Capabilities and Critical Capability Areas also appear on a block chart for each Functional Area. The detailed model criteria and questions are located together in a convenient tabular format in chapter 5.

3.1 Program Management (figure 3-1)

CCA 1.1, Management Authority, Responsibility, and Accountability. This CCA evaluates the offeror's organizational structure and control processes. Assignments of responsibility, span of control, and the interrelationship among software engineering and program management and system engineering are evaluated. This Critical Capability Area encompasses the following Critical Capabilities:

- 1.1.1 Organizational Approach
- 1.1.2 Management Control

CCA 1.2, Program Planning and Tracking. This CCA evaluates four offeror processes: program planning, contract work breakdown structure development, work package definition, and program schedule definition. The correlation between these planning and tracking processes is also evaluated. This Critical Capability Area encompasses the following Critical Capabilities:

- 1.2.1 Planning
- 1.2.2 Contract Work Breakdown Structure
- 1.2.3 Work Packages
- 1.2.4 Schedules

CCA 1.3, Subcontractor Management. This CCA evaluates the offeror's overall process to control, status, and report subcontractor development efforts. Issues of subcontractor capability evaluation, development management, and planning are evaluated. In particular, this CCA includes the flowdown of development requirements through the Systems Engineering Management Plan, Systems Engineering Master Schedule, Systems Engineering Detailed Schedule (SEDS), reviews, test, integration, and software development planning. The integration of the subcontractor activities with the prime is also evaluated. This Critical Capability Area encompasses the following Critical Capabilities:

- 1.3.1 Capability Evaluation
- 1.3.2 Subcontractor Development Management
- 1.3.3 Subcontractor Planning
- 1.3.4 Subcontractor Configuration Management

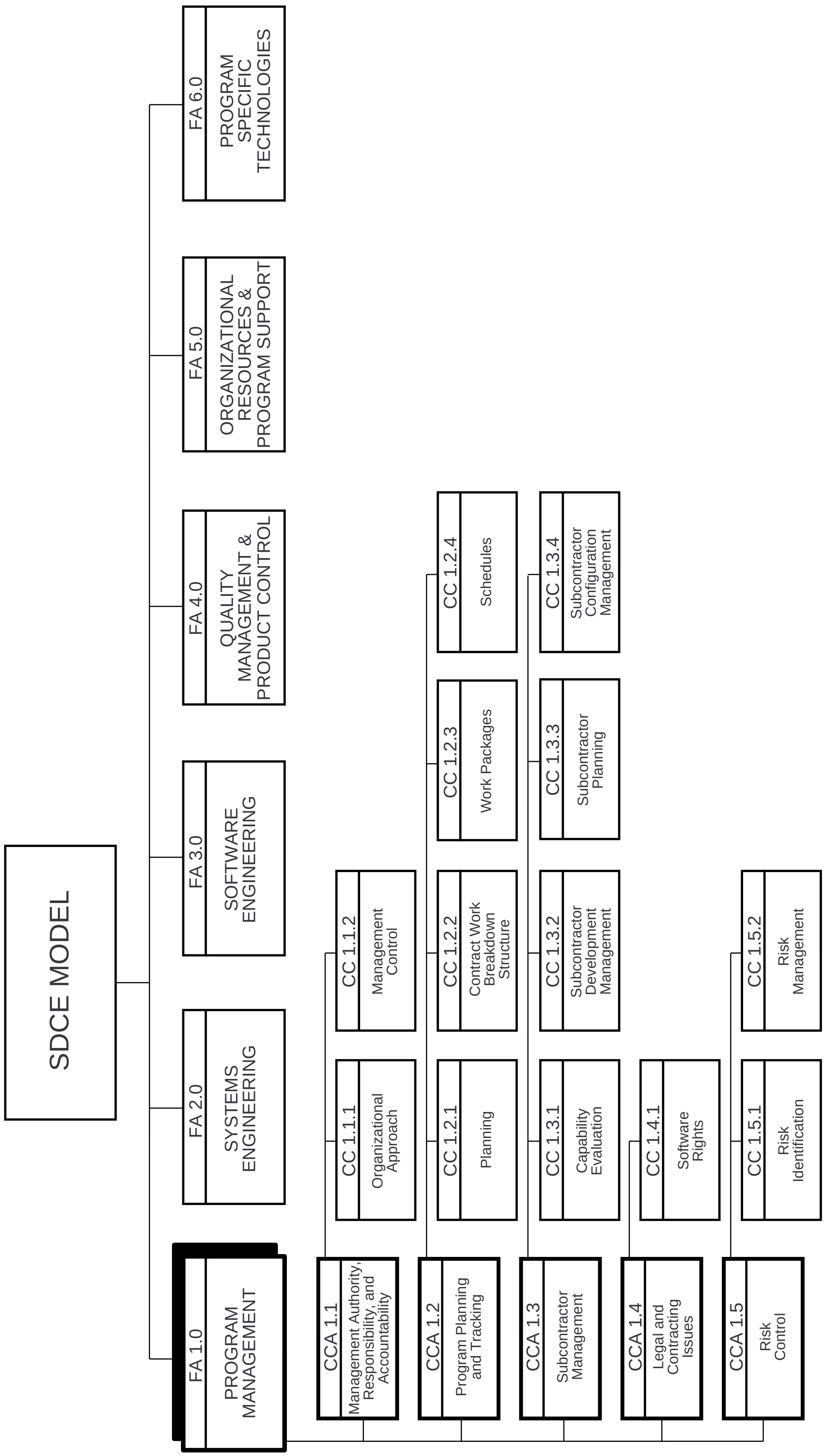


Figure 3-1. Program Management Block Chart

CCA 1.4, Legal and Contracting Issues. This CCA evaluates the offeror's process for identifying proprietary and restricted rights software and for establishing the necessary capability to develop and support the software given the restricted rights constraints. This Critical Capability Area encompasses the following Critical Capability:

1.4.1 Software Rights

CCA 1.5, Risk Control. This CCA evaluates the offeror's process for identifying and managing program risk. This Critical Capability Area encompasses the following Critical Capabilities:

1.5.1 Risk Identification
1.5.2 Risk Management

3.2 Systems Engineering (figure 3-2)

CCA 2.1, System Requirements Development, Management, and Control. This CCA addresses the development and allocation of system-level requirements, the adequacy of the requirements, the process by which changes to requirements are managed, the inclusion of a software perspective in system-level studies and reviews, and traceability from the system requirements to the requirements allocated to software. There may be multiple tiers between the highest-level system requirements and the level at which requirements are allocated to hardware and software. This Critical Capability Area encompasses the following Critical Capabilities:

2.1.1 Development and Allocation of Requirements
2.1.2 Adequacy of Requirements
2.1.3 Requirements Change Control
2.1.4 Software Impact Analysis
2.1.5 Requirements Traceability

CCA 2.2, Computer System Architecture Design and Review Process. This CCA addresses the definition and adequacy of the system-level architecture design (including hardware and software), system architecture design reviews, and architecture change impact analysis. This Critical Capability Area encompasses the following Critical Capabilities:

2.2.1 Architecture Definition
2.2.2 Adequacy of Architecture Design
2.2.3 Architecture Design Review
2.2.4 Architecture Change Analysis

CCA 2.3, Supportability. This CCA addresses reliability and maintainability issues, which are of concern to the support organization. This Critical Capability Area encompasses the following Critical Capabilities:

2.3.1 Reliability
2.3.2 Maintainability

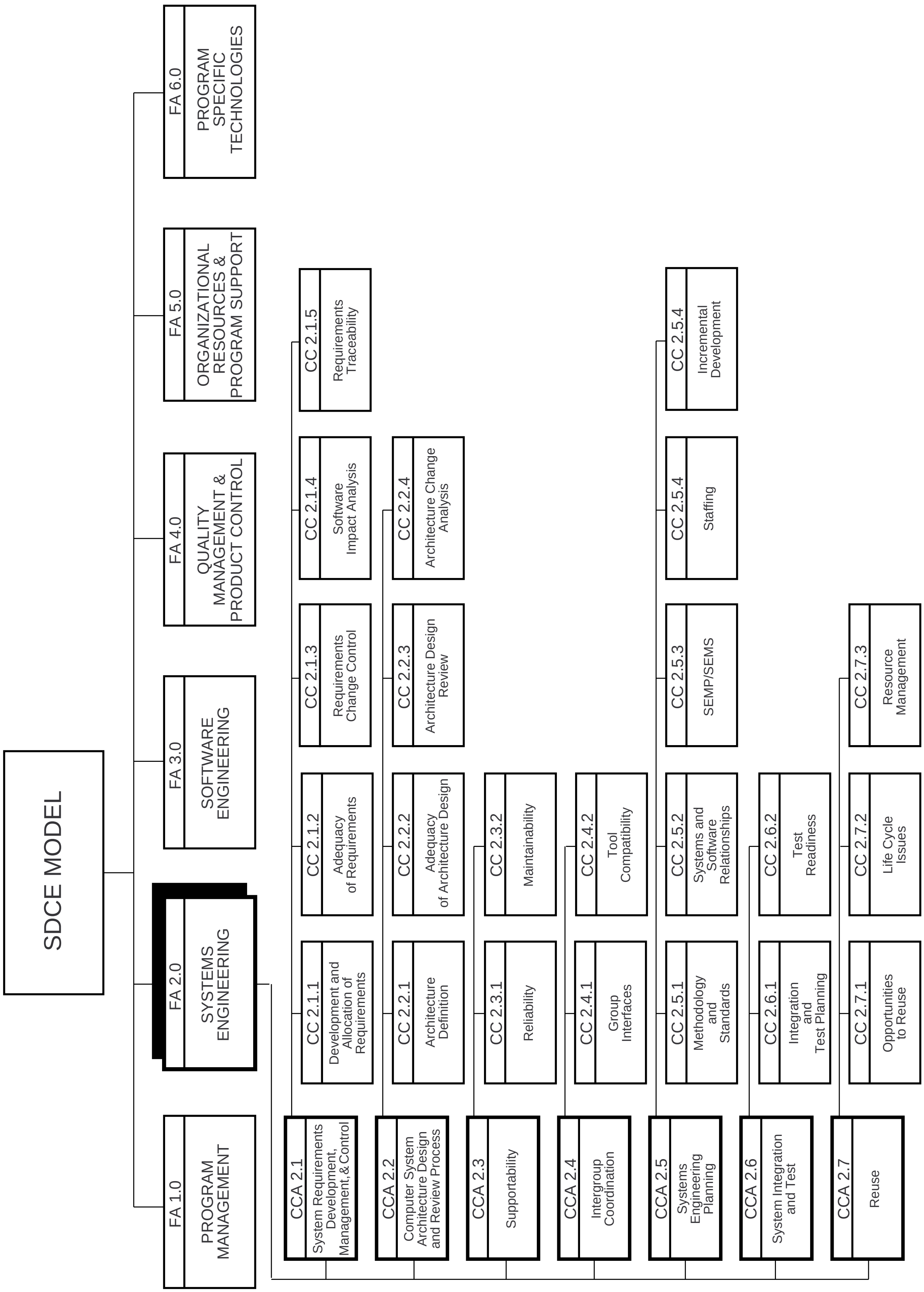


Figure 3-2. Systems Engineering Block Chart

CCA 2.4, Intergroup Coordination. This CCA addresses both coordination issues across different development groups, as well as coordination among developer, customers, users, and testers. This Critical Capability Area encompasses the following Critical Capabilities:

2.4.1 Group Interfaces
2.4.2 Tool Compatibility

CCA 2.5, Systems Engineering Planning. This CCA addresses the definition of systems engineering methods; their coordination with software engineering methods; the adequacy of the Systems Engineering Master Plan, Systems Engineering Master Shedule, and Systems Engineering Detailed Schedule; staffing; and incremental development plans. This Critical Capability Area encompasses the following Critical Capabilities:

2.5.1 Methodology and Standards
2.5.2 Systems and Software Relationship
2.5.3 SEMP/SEMS
2.5.4 Staffing
2.5.5 Incremental Development

CCA 2.6, System Integration and Test. This CCA addresses test planning, the adequacy of tools and facilities, and test readiness. This Critical Capability Area encompasses the following Critical Capabilities:

2.6.1 Integration and Test Planning
2.6.2 Test Readiness

CCA 2.7, Reuse. This CCA addresses opportunities to reuse existing components, to develop common components, and to develop new components with increased reuse potential, as well as the management of reusable resources. This Critical Capability Area encompasses the following Critical Capabilities:

2.7.1 Opportunities to Reuse
2.7.2 Life Cycle Issues
2.7.3 Resource Management

3.3 Software Engineering (figure 3-3)

CCA 3.1, **Software Development Planning.** This CCA ensures that the effort and resources required for meeting all of the requirements are planned for and devoted to the successful completion of the program. This Critical Capability Area encompasses the following Critical Capabilities:

3.1.1 Software Estimating
3.1.2 Software Work Packages
3.1.3 Software Engineering Development Methods
3.1.4 Preparing the Software Development Plan

CCA 3.2, Software Project Tracking and Reporting. This CCA ensures that program and engineering management stays informed on the status of each software component and the program

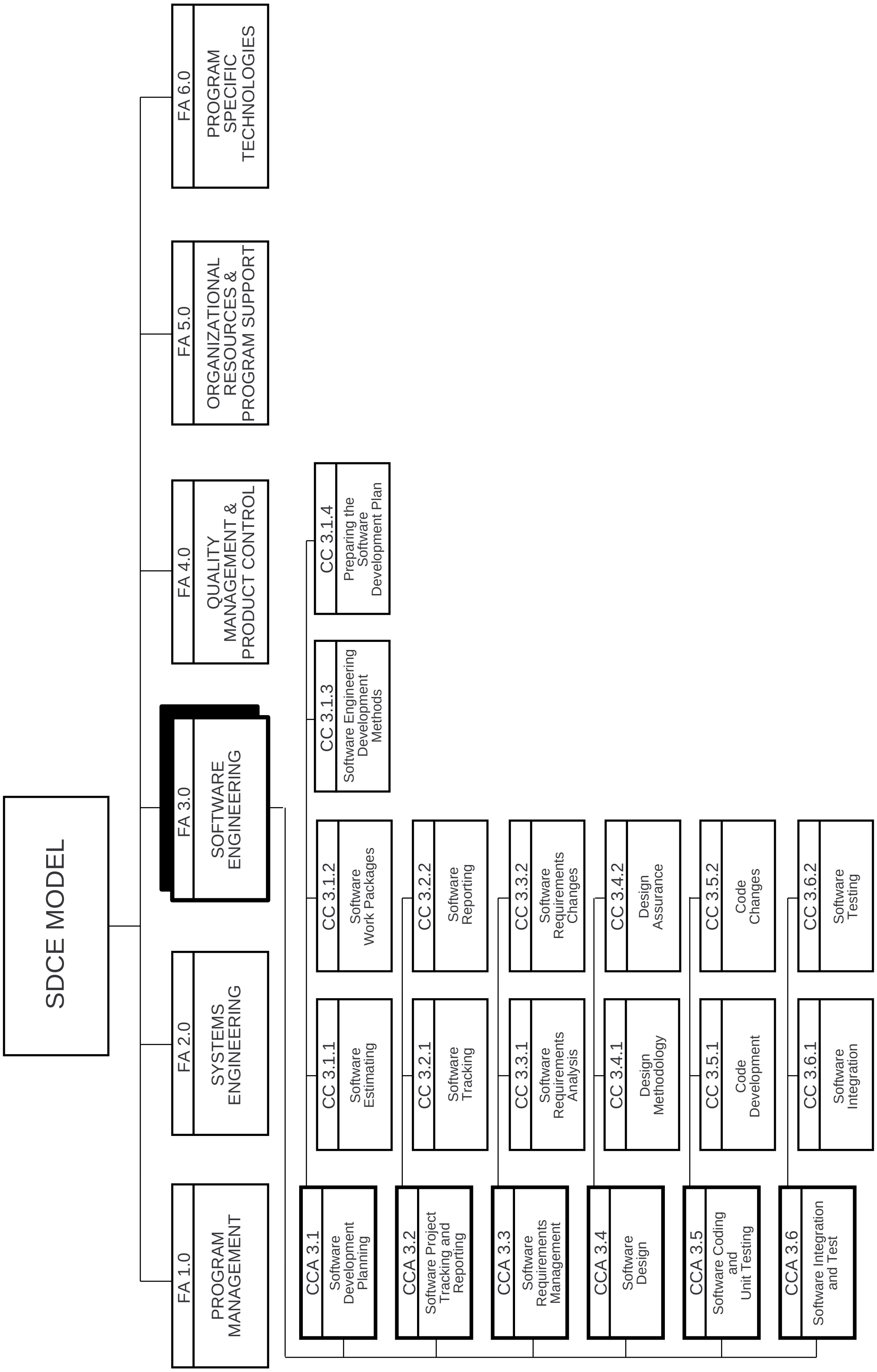


Figure 3-3. Software Engineering Block Chart

as a whole, and that corrective actions are taken when necessary. This Critical Capability Area encompasses the following Critical Capabilities:

- 3.2.1 Software Tracking
- 3.2.2 Software Reporting

CCA 3.3, Software Requirements Management. This CCA evaluates the processes used to analyze, use, and maintain the software requirements after they have been baselined. (The initial development of the software requirements is covered in the Systems Engineering Functional Area.) This Critical Capability Area encompasses the following Critical Capabilities:

- 3.3.1 Software Requirements Analysis
- 3.3.2 Software Requirement Changes

CCA 3.4, Software Design. This CCA evaluates methodologies and assurance mechanisms used to develop, document, and maintain the software design. This Critical Capability Area encompasses the following Critical Capabilities:

- 3.4.1 Design Methodology
- 3.4.2 Design Assurance

CCA 3.5, Software Coding and Unit Testing. This CCA evaluates the processes used to develop the object code and to perform the first-level testing, also known as component testing or unit testing. This Critical Capability Area encompasses the following Critical Capabilities:

- 3.5.1 Code Development
- 3.5.2 Code Changes

CCA 3.6, Software Integration and Test. This CCA evaluates the processes used to integrate the various software components and test the integrated components, sometimes referred to as blocks or builds. This Critical Capability Area encompasses the following Critical Capabilities:

- 3.6.1 Software Integration
- 3.6.2 Software Testing

3.4 Quality Management and Product Control (figure 3-4)

CCA 4.1, Software Quality Management. This CCA determines the quality of the program's software products and processes. This involves defining quality goals; establishing plans to achieve these goals; and monitoring and adjusting the software plans, activities, and goals to satisfy the needs of the customer and end user. This Critical Capability Area encompasses the following Critical Capabilities:

- 4.1.1 Quality Planning
- 4.1.2 Product Evaluations
- 4.1.3 Software Discrepancies

CCA 4.2, Software Quality Assurance (SQA). This CCA determines the existence of an organization whose functions are to ensure that the program standards are adhered to and the quality goals are met, to report quality findings to the development organizations and the program office, and

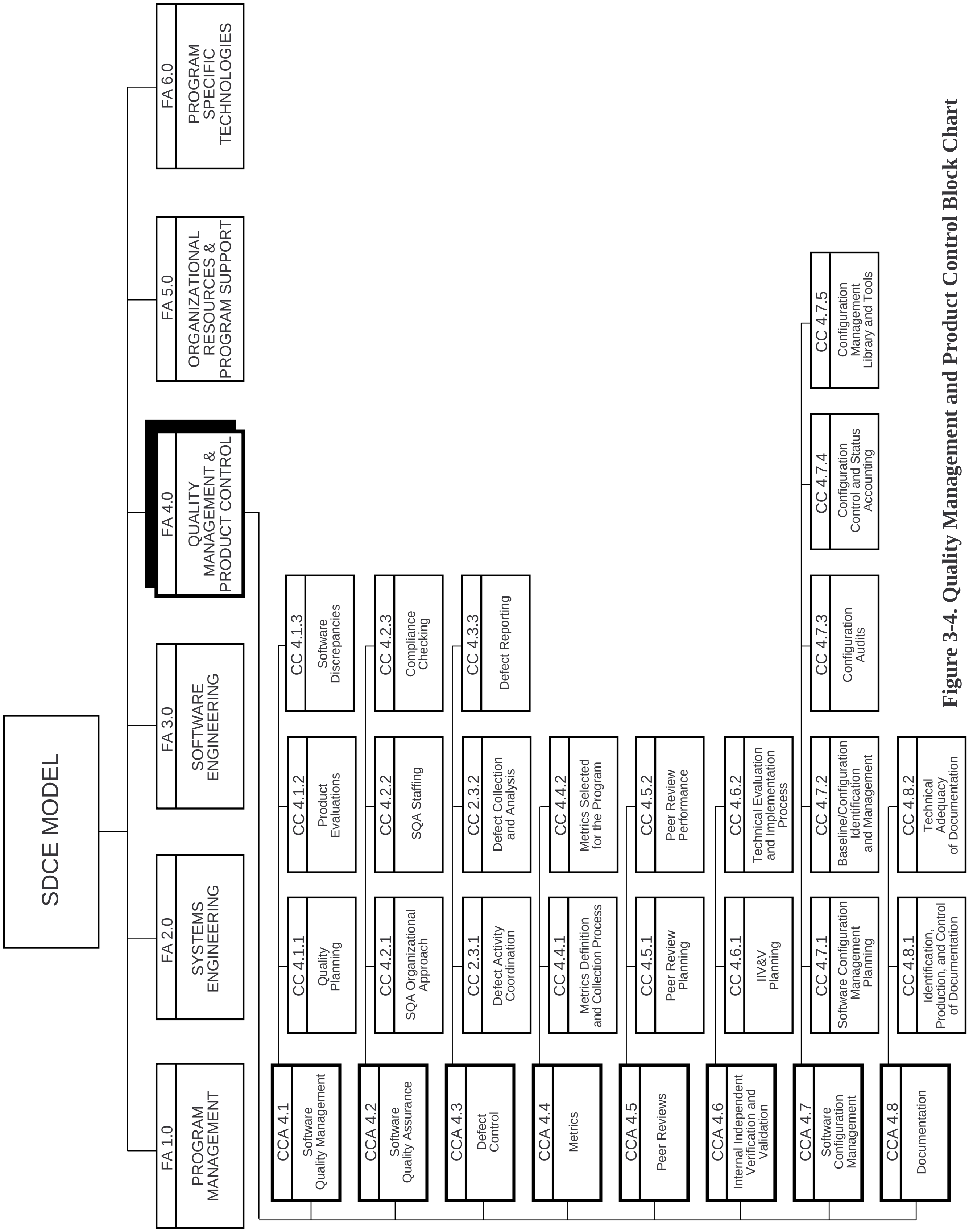


Figure 3-4. Quality Management and Product Control Block Chart

to elevate unresolved quality problems to management levels above the program. This Critical Capability Area encompasses the following Critical Capabilities:

4.2.1 SQA Organizational Approach
4.2.2 SQA Staffing
4.2.3 Compliance Checking

CCA 4.3, Defect Control. This CCA identifies the cause of defects and prevents them from recurring. Defect prevention involves analyzing defects that were encountered in the past and taking specific actions to prevent the occurrence of those types of defects in the future. This Critical Capability Area encompasses the following Critical Capabilities:

4.3.1 Defect Activity Coordination
4.3.2 Defect Collection and Analysis
4.3.3 Defect Reporting

CCA 4.4, Metrics. This CCA evaluates the bidder's capability to assess quantitatively the health status of the software and system development and management actives, and the bidder's capability to report its metrics results internally and to the customer, consistently throughout the development life cycle and across the different members of the bidding team. This Critical Capability Area encompasses the following Critical Capabilities:

4.4.1 Metrics Definition and Collection Process
4.4.2 Metrics Selected for the Program

CCA 4.5, Peer Reviews. This CCA develops a better understanding of the software products and removes defects early and efficiently. This involves implementing a set of pre-planned, methodical examinations of software products by the producers' peers to identify defects and areas where changes are needed. This Critical Capability Area encompasses the following Critical Capabilities:

4.5.1 Peer Review Planning
4.5.2 Peer Review Performance

CCA 4.6, Internal Independent Verification and Validation. This CCA assures that the critical elements of software undergo internal independent software verification and validation. Included is the assurance that the schedule will accommodate all the activities required for functional, performance, and documentation verification and validation. This Critical Capability Area encompasses the following Critical Capabilities:

4.6.1 IIV&V Planning
4.6.2 Technical Evaluation and Implementation Process

CCA 4.7, **Software Configuration Management.** This CCA establishes and maintains the integrity of the software products throughout the program's life cycle. This involves identifying the software configuration, systematically controlling changes to the configuration, and maintaining the integrity and traceability of the configuration throughout the life cycle. This Critical Capability Area encompasses the following Critical Capabilities:

4.7.1 Software Configuration Management (SCM) Planning
4.7.2 Baseline/Configuration Identification and Management
4.7.3 Configuration Audits

4.7.4 Configuration Control and Status Accounting
4.7.5 Configuration Management Library and Tools

CCA 4.8, Documentation. This CCA assures that documentation needed to perform the software engineering tasks (e.g., software requirements documents, software design documents, operation and maintenance manuals, test plans, and test procedures) is developed and reviewed. This Critical Capability Area encompasses the following Critical Capabilities:
4.8.1 Identification, Production, and Control of Documentation
4.8.2 Technical Adequacy of Documentation

3.5 Organizational Resources and Program Support (figure 3-5)

CCA 5.1, Organizational Standards and Procedures. This CCA develops and maintains a usable set of organizational policies, standards, procedures, and other process assets that provide programs with effective processes to use. These organizational policies, standards, and procedures and other assets are updated to reflect what programs have learned in using them. This Critical Capability Area encompasses the following Critical Capabilities:
5.1.1 System and Software Development Processes
5.1.2 Tailoring
5.1.3 Capturing and Making Available Use Information

CCA 5.2, Facilities. This CCA ensures that the facilities needed to perform the system and software development functions are planned in sufficient numbers and in accordance with the technical needs of the program. This Critical Capability Area encompasses the following Critical Capabilities:
5.2.1 Development Facilities
5.2.2 Specialized Facilities

CCA 5.3, Training. This CCA develops the skills and knowledge base of individuals so they can perform their roles effectively and efficiently. This Critical Capability Area encompasses the following Critical Capabilities:
5.3.1 Training Plans
5.3.2 Training Records and Effectiveness
5.3.3 Training Requirements

CCA 5.4, Human Resources. This CCA ensures that human resources are available to the program in sufficient numbers, that their allocation to the different functions and tasks meets the needs of the program, and that changes in their availability and allocation will not perturb the program. This Critical Capability Area encompasses the following Critical Capabilities:
5.4.1 Manpower Allocation Process
5.4.2 Manpower Availability and Retention

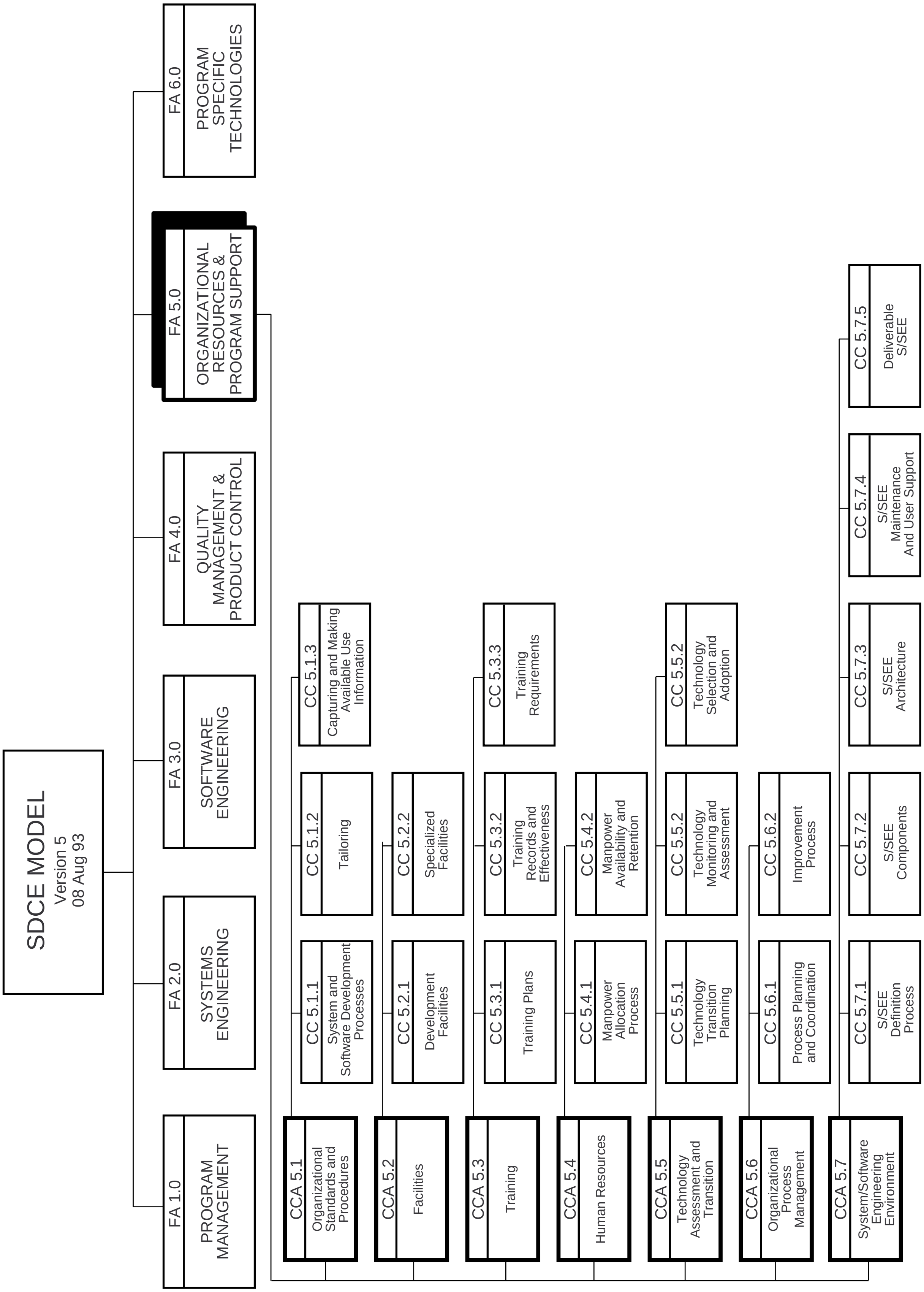


Figure 3-5. Organizational Resources and Program Support Block Chart

CCA 5.5, Technology Assessment and Transition. This CCA identifies and assess new technologies (i.e., tools, methods, and processes) and transition them into use in an orderly manner. This Critical Capability Area encompasses the following Critical Capabilities:

- 5.5.1 Technology Transition Planning
- 5.5.2 Technology Monitoring and Assessment
- 5.5.3 Technology Selection and Adoption

CCA 5.6, Organizational Process Management. This CCA coordinates the definition, use, and improvement of the processes used, with the intent of improving quality, increasing productivity, and decreasing the cycle time for product development. This Critical Capability Area encompasses the following Critical Capabilities:

- 5.6.1 Process Planning and Coordination
- 5.6.2 Improvement Process

CCA 5.7, System/Software Engineering Environment (S/SEE). This CCA ensures the availability to the program of an integrated set of software development tools which support the different development and management functions, is consistent with the processes and methodologies and languages selected for the program, and will be available during the development and support phases of the program. This Critical Capability Area encompasses the following Critical Capabilities:

- 5.7.1 S/SEE Definition Process
- 5.7.2 S/SEE Components
- 5.7.3 S/SEE Architecture
- 5.7.4 S/SEE Maintenance and User Support
- 5.7.5 Deliverable S/SEE

3.6 Program Specific Technologies (figure 3-6)

CCA 6.1, Artificial Intelligence (AI). This CCA evaluates the offeror's experience and expertise in applying AI tools and techniques to software development. AI projects can be divided into two parts: the task of domain, which describes the problem to be solved, and the technology used, which describes the software methods used to solve the problem. For example, natural language translation, reasoning by analogy, and fault diagnosis are task domains, while neural networks, case-based reasoning, and nonmonotonic logic are technologies.

- 6.1.1 AI Task Domain Analysis
- 6.1.2 AI Tools and Technology
- 6.1.3 Specific AI Technology
- 6.1.4 AI Management Process
- 6.1.5 AI Development Process
- 6.1.6 Personnel Skills And Qualifications for AI
- 6.1.7 AI Capability Demonstrations and Risk Management

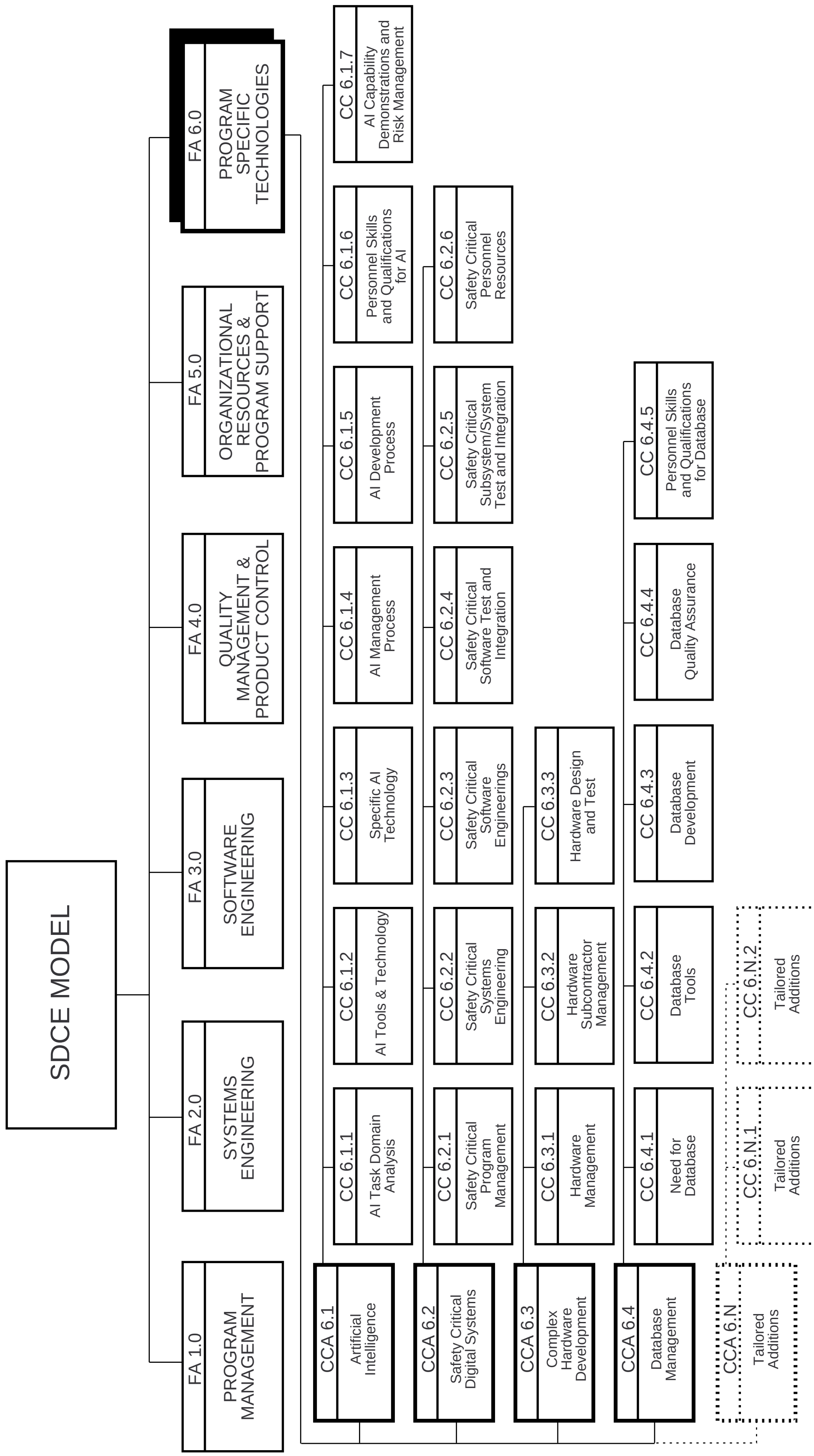


Figure 3-6. Program Specific Technologies Block Chart

CCA 6.2, Safety Critical Digital Systems. This CCA evaluates the offeror's capability and capacity to develop systems containing Safety Critical Digital Systems. This Critical Capability Area encompasses the following Critical Capabilities:

- 6.2.1 Safety Critical Program Management
- 6.2.2 Safety Critical Systems Engineering
- 6.2.3 Safety Critical Software Engineering
- 6.2.4 Safety Critical Software Test and Integration
- 6.2.5 Safety Critical Subsystem/System Test and Integration
- 6.2.6 Safety Critical Personnel Resources

CCA 6.3, Complex Hardware Development. This CCA evaluates the offeror's processes and procedures for managing and developing complex custom integrated circuits. This Critical Capability Area encompasses the following Critical Capabilities:

- 6.3.1 Hardware Management
- 6.3.2 Hardware Subcontractor Management
- 6.3.3 Hardware Design and Test

CCA 6.4, Database Management. This CCA evaluates the offeror's capability in applying software development processes and procedures to the development of large databases. This Critical Capability Area encompasses the following Critical Capabilities:

- 6.4.1 Need for Database
- 6.4.2 Database Tools
- 6.4.3 Database Development
- 6.4.4 Database Quality Assurance
- 6.4.5 Personnel Skills and Qualifications for Database

CHAPTER 4. DESCRIPTION OF THE SDCE PROCESS

Applying the SDCE process to a particular source selection involves thirteen major activities that are presented in the activity flow and summaries in subsection 2.2. The activity flow diagram (figure 2-6) establishes an approximate sequence for the tasks; however, many things can be done in parallel, and the exact order of some of the tasks can be arranged to best suit the schedule at hand. In particular, the evaluation, scoring, and integration tasks described in section 4.I are collected to promote an understanding of these tasks as a functionally unified activity; in practice, these activities are performed integrally with the activities described in sections 4.F (review proposals), 4.G (plan for and conduct site visit), 4.H (analyze CRs and DRs), and 4.J (incorporate into contract).

Detailed descriptions of the thirteen SDCE activities are contained in sections 4.A through 4.M that correspond exactly to the flow blocks in figure 2-6. Each section begins with an overview and a diagram (based on the more detailed task flow in figure 2-7) setting its tasks in the context of the overall SDCE process. The remainder of the section contains detailed instructions, guidelines, and observations on the tasks to be performed for that activity.

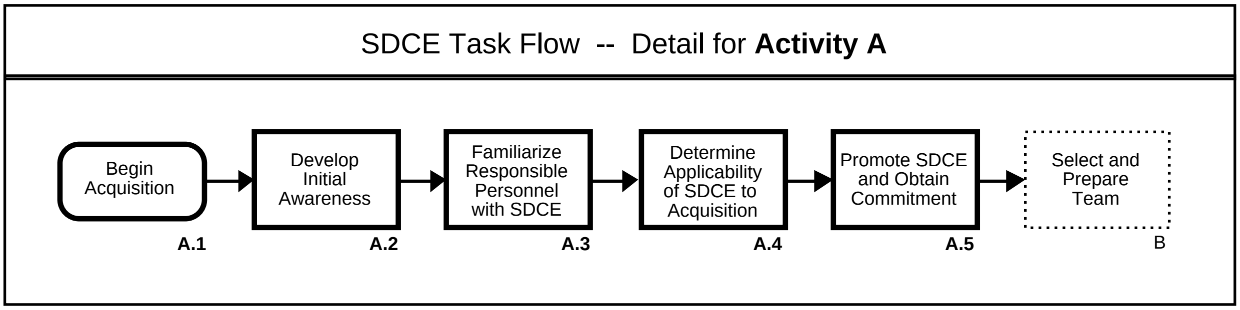
Throughout the SDCE planning and execution process, it is important to be consistent in using the method. Source selection rules require consistent evaluation of all offerors. Table 4-1 lists guidelines to help maintain this consistency.

Volume 2, attachment 3-10 contains the SDCE Team Activity Worksheet/Checklist. This checklist that can be used to tally the various items that must be completed during the course of applying the SDCE method to a particular source selection. The checklist is organized around the thirteen major activities.

Table 4-1. Consistency Guidelines

- All applications of the SDCE in source selections should be conducted in accordance with the SDCE policy and published SDCE method description.
- The SDCE team should be chaired by a software-experienced senior acquisition engineering professional.
- Whenever possible, a single core team should do all the site visits. Members of the core SDCE team should be fully qualified in the source selection process and the SDCE method.
- When selecting elements of the model to be applied on a source selection, the model elements (FAs, CCAs, and CCs) should be used without modification to the maximum extent applicable.
- The method should be tailored by senior, experienced acquisition professionals, including program stakeholders.
- The tailored SDCE question set to be used should be released with the program RFP.
- The SDCE team should all be experienced in source selection, and experienced and trained in the SDCE method.
- Program offices planning to use the SDCE method should contact their local AFMC product or logistics Center SDCE OPR for support in planning and conducting the SDCE. The SDCE advisor should help to ensure consistency.
- Site visit feedback should be accomplished consistently with the descriptions in this SDCE pamphlet and AFFARS, Appendix AA.

Section 4.A Determine Applicability



The initial activity in using the SDCE method on a given program is to determine its applicability to that program. This section provides the following guidance to the acquisition and support organizations in determining the applicability and performing preplanning activities:

- *How to make the program office aware of SDCE*
- *How to familiarize the program office with SDCE method*
- *How to determine whether to apply SDCE*
- *How to promote the use of SDCE*
- *How to obtain a commitment to use SDCE*

4.A.1 Begin Acquisition

Planning for the source selection begins when the program management directive is released. Planning for the application of the SDCE method should begin as early in this planning phase as possible to fully integrate the SDCE method into the source selection process.

4.A.2 Develop Initial Awareness

Program offices need to be aware of the SDCE method to ensure that it is considered for use early in the procurement process. As soon as a new program is starting its acquisition planning for a source selection in either the Demonstration/Validation phase or the Engineering and Manufacturing Development phase, communication should begin between the program office and the Center SDCE OPR.

Both the Center SDCE OPR and the program office have a responsibility to initiate communication at the earliest possible time. The Center SDCE OPR should be aware of all new program starts in order to be able to advise program offices early in the acquisition planning phase concerning the potential use of the SDCE method. Conversely, the program office has a similar responsibility to initiate this early communication.

Once communication has begun, the Center SDCE OPR should provide an SDCE executive overview briefing to the principal program office management, procurement, and engineering leaders to make them aware of the SDCE policy, method, and risk reduction support to their pending source selection.

This briefing would provide an opportunity for program office officials to raise questions concerning the SDCE method and its applicability to the pending source selection. A briefing outline appropriate for this overview is shown in table 4-2. The referenced charts are located in Volume 2, attachment 4. The Center SDCE OPR should review the initial planning documents and meet with additional program office personnel to initiate the applicability determination. The documents to review might include, for example: the operational requirements document, program management directive, acquisition program plan, system specification, and source selection plan.

Table 4-2. SDCE Executive Overview Briefing Outline

Vol 2, page	54	SDCE Title/Logo/Pictures
	55	Outline
	56	SDCE Overview
	57	Background
	59	SDCE Approach
	60	SDCE Role in SE/CM Process
	61	SDCE Role in Source Selction
	64	SDCE Development
	65	SDCE Model Structure
	66	SDCE Functional Areas
	67	Critical Capability Areas
	68	Critical Capability Areas (Continued)
	69	Critical Capability Areas (Continued)
	78	SDCE Questions
	80	SDCE Activity Flow
	87	SDCE Guidebook Contents
	89	Sample SDCE Plan - Cover and Contents
	90	Example of Incorporating SDCE into Source Selection Structure
	94	SDCE Proposal Data
	103	"No Discussions" Impact
	104	Features
	105	Features (Continued)
	106	Policy on Evaluating Contractor's Capability
	108	SDCE Summary

In addition to reviewing these documents, the Center SDCE OPR should meet with the responsible program systems engineering and software personnel to discuss program characteristics affecting SDCE applicability.

4.A.3 Familiarize Responsible Program Office Personnel with SDCE

Once the system programs office (SPO) leadership has been briefed on the SDCE, the Center SDCE OPR should familiarize other key SPO stakeholders on the specifics of the SDCE method. The purpose of this SDCE familiarization briefing would be to explain what the SDCE method is, why it would be helpful in reducing program risk, how it would be applied to the program, when the site visits would be accomplished in the context of the source selection process, and who would be involved and how much effort would be required in conducting the site visits. The briefing outline in table 4-3 should be tailored as much as possible to the acquisition at hand. The referenced charts are located in Volume 2, attachment 4.

The responsible SPO systems engineers, software engineers, and contracting officers should be involved in this SDCE familiarization task. It is advisable to establish a group within the SPO to gather the necessary information and work issues to determine whether to apply the SDCE to this acquisition. This group typically would include a program manager, senior program/project engineer, senior software engineer, and the procuring contracting officer (PCO). The group would make a recommendation on the applicability of SDCE to the SPO director. If the director decides to use the SDCE method, the group would be the logical people to follow through in working initial issues and planning to establish an SDCE evaluation team to apply the SDCE method.

4.A.4 Determine Applicability of SDCE to Acquisition

Applicability is determined by considering a combination of factors, including policy on use of the SDCE, application guidelines, the program acquisition strategy, and the characteristics and needs of the particular program.

Use Application Guidelines. Since the SDCE method is intended to reduce the risk in successfully executing a development program, the application decision is based on anticipated program risks. Table 4-4 lists program characteristics related to development risks.

Review Acquisition Strategy. Given the application policy and guidelines, it is important to review the program acquisition strategy for additional insight into whether to apply, and how to apply, the SDCE method. As an example, if the acquisition strategy is to award a sole source contract, it might still be advisable to review the offeror’s software development capability and capacity using the SDCE method. Further, if new software technology, such as Ada, is required for the sole source development, it would be advisable to evaluate the offeror’s capability to develop software in Ada. Characteristics of the acquisition strategy, such as multiphase development, incremental development, head-to-head competitive development, down selecting from Dem/Val into EMD, and numerous other acquisition strategies could impact the decision to apply the SDCE method.

Develop Recommendation on SDCE Use. Once all factors have been considered, the Center SDCE OPR, together with the responsible SPO participants, should prepare a recommendation on the use of the SDCE method for the program. This recommendation should be based upon the policy, guidelines, acquisition strategy, and needs of the program going into source selection. This recommendation should be coordinated with SPO engineering, management, and contracts.

Table 4-3. SDCE Program Office Familiarization Briefing Outline

Vol 2, page	54	SDCE Title/Logo/Pictures
	56	SDCE Overview
	57	Background
	59	SDCE Approach
	60	SDCE Role in SE/CM Process
	61	SDCE Role in Source Selection
	62	Placement of SDCE within Source Selection Structure
	63	Findings from Data Gathering
	64	SDCE Development
	65	SDCE Model Structure
	67	Critical Capability Areas
	68	Critical Capability Areas (Continued)
	69	Critical Capability Areas (Continued)
	70	SDCE Model Block Chart
	71	Program Management Block Chart
	72	Systems Engineering Block Chart
	73	Software Engineering Block Chart
	74	Quality Management and Product Control Block Chart
	75	Organizational Resources and Program Support Block Chart
	76	Program Specific Technologies Block Chart
	77	The SDCE Model
	78	SDCE Questions
	79	Example Format for Criteria and Questions
	80	SDCE Activity Flow
	81	SDCE Activities - Preparation
	82	SDCE Activities - Preparation (Continued)
	83	SDCE Activities - Conduct
	84	SDCE Activities - Conduct (Continued)
	85	SDCE Activities - Wrap-up
	86	Team Makeup
	87	SDCE Guidebook Contents
	88	SDCE Products
	89	Sample SDCE Plan - Cover and Contents
	90	Example of Incorporating SDCE into Source Selection Structure
	91	SDCE Schedule Integrated with Source Selection Key Events
	92	SDCE Schedule Integrated with Source Selection Key Events (Continued)
	93	SDCE Site Visit Schedule Template
	94	SDCE Proposal Data
	95	Example Cover Sheet for Project Sample Data
	96	Example Capability Definition Matrix
	97	Example Capability Implementation Matrix Showing Project Sample Data
	98	Example Capability Implementation Matrix Showing Integration Problem Areas
	99	Example Capability Evaluation Matrix
	100	Relationship of Offeror-Completed Forms to Capability Evaluation Matrix
	101	SDCE Roll-up Process
	102	Costs
	103	"No Discussions" Impact
	104	Features
	105	Features (Continued)
	106	Policy on Evaluating Contractor's Capability
	107	Center OPR
	108	SDCE Summary

Table 4-4. When to Apply SDCE

The SDCE process should be applied on weapon system EMD phase source selections when two or more of the following conditions, requirements, or characteristics exist. Even if only one exists, applying the SDCE may be appropriate for the particular acquisition. • The development program is a major (DAB/AFSARC review required) program. • The program software development is anticipated to cost more than $25M or require the development of more than 100K lines of code. • The program development involves highly complex requirements and associated complex software development. • A complex software/systems integration effort is expected. • The software development is constrained by an aggressive program schedule or it is anticipated that the software development will be on the critical path. • The program development involves safety critical software (e.g., human safety and nuclear surety factors). • The program or its software development includes unprecedented functional capabilities or is likely to employ significant new software technologies (e.g., language or design methodologies). • It is anticipated that there might be bidders with uncertain software development and management capabilities or unknown experience with the program application domains. • The program is software intensive (i.e., the functionality of the system is primarily contained in the software). The SDCE process should be considered on weapon system Dem/Val source selections when either of the following conditions exists. • Significant software is planned to be reused in the EMD phase. • Major EMD software contractors are planned to be downselected from among the Dem/Val phase contractors.

4.A.5 Promote SDCE and Obtain Commitment to Use

Following the initial recommendation to apply the SDCE on the program, it is important to promote and explain the SDCE method throughout the SPO to assure all participants are aware of the SDCE method and its potential risk reduction support to their disciplines. The disciplines and associated participants would typically include program/project management, systems engineering, software engineering, integration and test, contracts/procurement, configuration and data management, financial management, program control, and logistics.

Communicate Costs and Benefits. In explaining the resources required to conduct an SDCE, it is useful to provide estimates of both the effort and costs associated with the method. Table 4-5 provides initial estimates for the SDCE method, based on experience with the SDCCR and other evaluation methods. Actual effort will vary, depending on the scope of the program.

Table 5-5. Effort Required to Conduct SDCE

Activity	Effort (Person Days)
SDCE team preparation through final RFP release	20– 80
SDCE team proposal analysis prior to site visit (per offeror)	18– 24
SDCE team site visit (per offeror), plus travel/support cost	24– 30
SDCE team evaluation (per offeror) after site visit	18– 24
Contractor preparation (first time SDCE)	200–400
Contractor site visit	20– 60
Contractor follow-on	15– 30

The risk reduction benefits of incorporating the SDCE method into accquisition planning and source selection must also be explained in terms of the potential benefits to the program. These benefits include:

- Identification of specific capabilities each offeror has in place to support successful software development.
- Offeror's commitment to apply existing capability and processes to the program.
- Resolution of proposal limitations, ambiguities, and conflicts through site visit dialogue.
- Insight by program office members of the SDCE team into the offerors' software development capability.

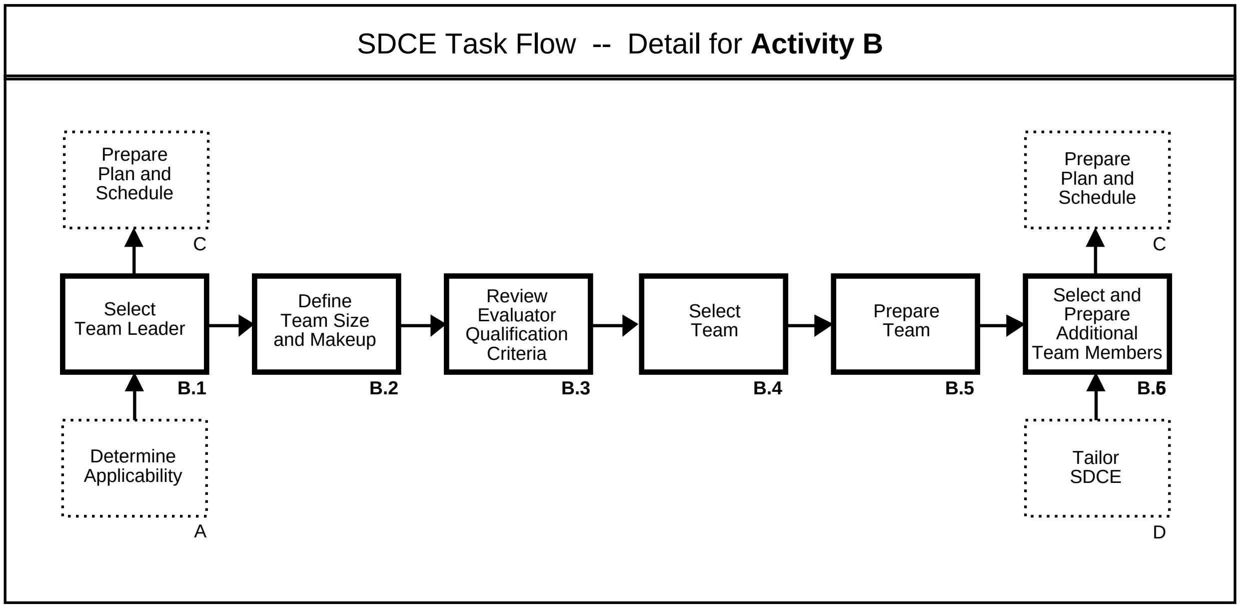
Communicate Current Policy. Policy on the use of the SDCE method should be explained to all responsible SPO participants. The challenge posed by the policy is that the criteria for determining when to apply the SDCE are stated in objective, and in some cases quantitative terms, such as dollar value of the program or its software development and number of source lines of code to be developed. Too often, these numbers are not known or at best are estimates during the early stages of program formulation. The real intent of the policy is to direct the application of the SDCE method when a program contains a significant software development effort. Thus, it is useful to explain that the SDCE method policy is intended to promote risk reduction in software development.

Provide SDCE Recommendation to SPO Director. Once the SDCE recommendation is developed, coordinated, and explained to SPO participants, it is forwarded to the SPO director. The recommendation could be in the form of a brief memo or a stand-up briefing. It should address the why, who, how, and when questions for a positive recommendation. Also, for a positive recommendation, it would be prudent to recommend a chairperson of the SDCE team, who would then follow through with the necessary planning, including SDCE team formation.

Obtain Commitment from Leaders. Following the decision of the SPO director to proceed with the SDCE, it is important to communicate with the leaders of the SPO to obtain their commitment and support in planning and conducting the SDCE. Depending on how the SPO is structured within the overall Center organization, it may also be helpful to brief up the chain of command from the program manager to the organizational directors responsible for the program. Also, the functional directors and chiefs should be briefed to solicit their commitment and support in conducting the SDCE. For example, it is essential to obtain solid support from the director of engineering as well as the chief engineer on the program. These senior engineering leaders will be providing engineering personnel to participate on the SDCE team and may have concerns and inputs to the conduct and tailoring of the SDCE model and process. They may, for example, want to be sure the requirements definition and allocation process is emphasized in conducting the SDCE.

Commitment from the functional organization directors is also required since they will be providing acquisition professionals to participate on the team. These might, for example, include engineering, financial management, logistics, procurement, and configuration and data management. This commitment will include identifying individuals to participate on the SDCE team.

Section 4.B Select and Prepare Team



The first activity in performing the SDCE is to identify the team that will conduct the evaluation. This section provides the following guidance to assist in the selection and preparation of the evaluation team:

– *How to select the team leader*

– *How to determine team size and makeup*

– *How to review evaluator qualification criteria*

– *How to select members for the evaluation team*

– *How to prepare the evaluation team*

4.B.1 Select Team Leader

The first task in forming the evaluation team is to select the team leader. This person should have as a minimum 15 years of acquisition and/or development experience, considerable systems and software engineering experience, the ability to lead small groups, and the ability to convincingly present the results of the evaluation. The team leader could be selected from within the program office or Center staff depending upon experience level, qualifications, and availability.

4.B.2 Define Team Size and Makeup

An evaluation team actually consists of two parts, a core team and a support team.

The core team should consist of the team leader plus two or three additional senior members. Based on the characteristics of the software development, technical requirements and factors, and the final

CCAs and CCs selected, the core team members should assure coverage of the following areas: (1) systems engineering, (2) software engineering, (3) program management, and (4) logistics engineering. The role of the core team is to provide technical expertise to evaluate the offerors' software development capability. Once constituted, the core team will stay together throughout the entire SDCE evaluation process, thereby providing the evaluation with stability and a cadre of highly experienced senior personnel. It is highly recommended that one or more of the core team, such as the team leader and/or the senior systems engineer, be members of the program office. This will assure representation from the program office (stakeholder) and assure the stakeholder has visibility into and a strong commitment to the entire evaluation process. Refer to paragraph 4.B.3 for suggested qualification criteria. The core team members are also members of the SSEB and are expected to fully participate in the evaluation of the proposals.

The support team supplements the core team with specific skills, knowledge, and experience required by the evaluation. Support team members may not be required to participate in the entire evaluation, but, depending on the size and complexity of the program, may be called upon to assist the core team in evaluating specific capability areas not covered by the core team expertise. Suggested support team disciplines include the following: (1) software engineering, (2) subsystem engineering, (3) contracting, (4) quality assurance, (5) configuration management, (6) test, (7) financial management, and (8) software management. Refer to paragraph 4.B.3 for suggested qualification criteria.

4.B.3 Review Evaluator Qualification Criteria

After the team leader has been selected, it is important to assure that the required evaluator qualifications are properly defined and that the given qualification criteria are applicable to this acquisition. In order to accomplish this at an early stage of source selection planning, the team leader should work closely with the program office in planning the overall source selection, thereby becoming familiar with the size and scope of the SDCE required.

Application domain experience of the program to be evaluated should be distributed across the core and support teams. Since no one person will be knowledgeable of the total system, this knowledge distribution will tend to assure that all aspects of the system are covered. Team members who lack basic knowledge of the application must be made familiar with it before the evaluation begins.

The following suggested evaluator qualification criteria will assure that all evaluation team members are properly qualified prior to the actual evaluation. Applicability of the criteria should be based on the complexity, technical structure, and assessment criteria of the program. This information should be available through the interaction of the team leader with the program office. The SDCE team should have a mix of talents consistent with the characteristics of the development program. Experienced professionals are required, with knowledge of the software development process, the systems engineering process, the technology to be implemented in the program, the program application area, program management, and the specific procurement. Core team members should each have a minimum of seven years of related experience; the support team should be made up of experienced personnel as available.

4.B.3.1 Core Team Qualification Criteria

Senior Systems Engineer. The senior systems engineer should have experience in the systems engineering process, such as: (1) experience in the transformation of validated customer needs and requirements into a life cycle-balanced solution set of system product and process designs , (2) requirements definition and specification, (3) SEMP and SEMS, and (4) systems engineering requirements in the RFP.

Senior Software Engineer. The senior software engineer should have an extensive background and understanding of the software development process as follows: (1) requirements analysis, (2) design, (3) code and test, (4) integration, (5) software engineering processes and procedures, (6) project planning and estimation, and (7) software engineering requirements in the RFP.

Senior Project Manager. The senior program manager should have experience and understanding of the fundamental concepts and principles of project planning, process models, project scheduling, and milestones. The project manager should also have experience in project organization and management issues, development team organization, project costing, and management requirements in the RFP.

S**enior Logistics Engineer.** The senior logistics engineer should have knowledge of all the activities related to the development and support of software systems. The logistics engineer should be familiar with the software product life cycle, software support environments, software documentation and training, and logistics requirements in the RFP.

4.B.3.2 Support Team Qualification Criteria

Software Engineering. The software engineer should have experience in all aspects and activities of the software development life cycle as follows: (1) requirements analysis, (2) design, (3) code, (4) test, and (5) integration. The software engineer should also be able to evaluate software products for conformance to standards and specifications, verify and validate software, and evaluate technologies and tools. The software engineer should also be familiar with fundamental programming concepts and operating systems.

Lead Project/Subsystem Engineering. Lead project/subsystem engineering representatives should have domain expertise and experience for the major subsystems of the system being acquired. The general qualifications of these team members should include acquisition engineering, subsystem engineering, software engineering, systems engineering, and system acquisition.

Contracting. The contracting support person should be familiar with the solicitation, analysis, evaluation, and negotiation of proposals for systems that are software intensive. This person should also be familiar with procurement planning, such as: (1) development of the acquisition strategy, (2) solicitation document preparation, (3) proposal evaluation, (4) review of cost analysis documentation, (5) competitive range determination, and (6) preparation of prenegotiation position. In addition, the contracts representative should have a background in negotiating individual contracts to obtain services for the government and preparing recommendations for contract award, and must be familiar with the contract requirements in the RFP.

Quality Assurance. The software quality assurance representative should have knowledge of software quality assurance plans and their contents; should know how a quality assurance program is initiated and conducted; and should have experience in identifying quality assurance key issues, staffing the quality assurance function, training, schedule development, and monitoring the quality assurance program. The software quality assurance representative must be familiar with the quality assurance requirements in the RFP.

Configuration Management. The configuration management support person should be familiar with the following configuration management activities: (1) configuration/baseline control, (2) change management and version control, (3) configuration control board operations, and (4) configuration management methods. Additionally, this person should be familiar with the establishment of a software configuration management system, including plans, objectives, responsibilities, and the approach and methods to be used. The configuration management representative must be familiar with the configuration management requirements in the RFP.

Test. The software test support person should be well versed in the basic types of software test methods, have a strong background in test planning and execution, and be familiar with the requirements in the RFP.

Financial Management. The financial management support person should be intimately familiar with cost estimating processes and the steps required to reliably establish a software cost estimation activity. These steps are as follows: (1) how to establish objectives, (2) how to plan for required data and resources, (3) how to use several independent techniques and sources, (4) how to compare and iterate estimates, and (5) how to collect actuals and compare to estimates. The financial management support person must be familiar with the system and software cost requirements of the RFP.

Software Management. The software manager should have experience in activities and with policies related to the development and support of mission critical software. This person should be able to identify critical risk areas associated with software development due to system complexity and have knowledge and experience with various aspects of software development throughout the planning and development life cycle. In addition the software manager must be familiar with the software management requirements in the RFP.

4.B.4 Select Team

After preliminary source selection planning has been accomplished and specific source selection parameters and conditions (such as discussions/no discussions, number of bidders, technical structure of the source selection, and assessment criteria) have been determined, the team leader and representatives from the program office should prepare a candidate list of core evaluation team members and, if necessary, a list of appropriate support team members. This list of evaluators should include personnel from the program office (stakeholders) as well as specialists from the local staff organization. Candidate team members should be selected on the basis of qualification criteria in paragraph 4.B.3 or any additional requirements that the team leader and program office representatives feel apply to the program to be evaluated. Also, it is highly recommended that firm commitments be received from the parent organizations of the evaluation team members.

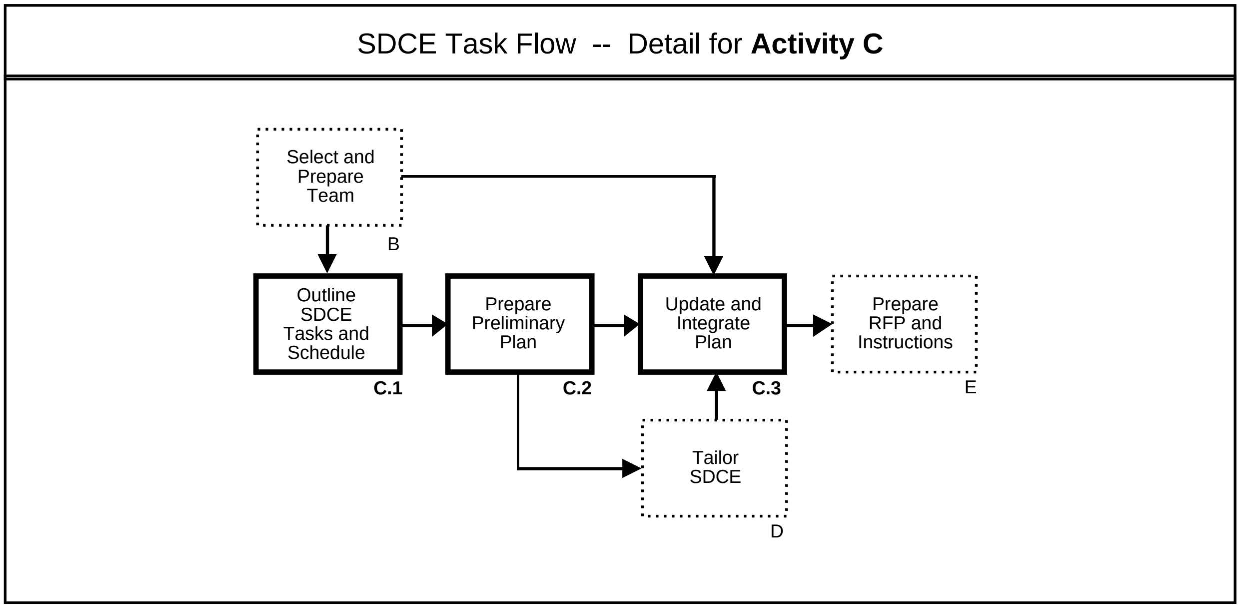
4.B.5 Prepare Team

The goal of SDCE team preparation is to ensure that team members have the skills and knowledge to perform the evaluation. The team leader or designated representative should ensure that the preparation needed by each team member is available and provided prior to participation in the SDCE. Members of the evaluation team who have been previously trained should participate in training the new team members. The evaluation team members must understand the organization and structure of the current source selection and must contribute to evaluation planning. A typical team preparation schedule may include but not be limited to the following: (1) meet with program management and engineering staff to get briefed into the program (familiarization with program office key issues); (2) review formal statement of operational requirements, e.g., statement of need; (3) review program management directive; (4) review acquisition strategy plan; (5) review RFP, including draft systems specification; (6) familiarize team with source selection philosophy; (7) train evaluation team in SDCE methodology; and (8) discuss the detailed evaluation plan, including the purpose of the site visits, topics for each site visit, team leader's role during site visits, participants, and their roles.

4.B.6 Select and Prepare Additional Team Members

Selection and preparation of additional SDCE team members is based on the results of the SDCE tailoring activity (section 4.D). Once the final set of CCAs and CCs has been selected, the SDCE team leader must determine if the core and support team members have all the necessary expertise to evaluate those CCAs and CCs. If not, the team leader should take action to select and prepare additional team members in accordance with paragraphs 4.B.4 and 4.B.5.

Section 4.C Prepare Plan and Schedule



Determining the SDCE schedule and preparing planning information for incorporation into the overall source selection plan are critical tasks that must be initiated as early as possible. This section outlines this planning process and provides guidelines for preparing necessary documents. Relevant SDCE information may be incorporated directly into the source selection plan and evaluation guide or may be defined in separate planning documents. In particular, this chapter provides guidance to help the SDCE team leader and the individual team members with the following tasks:

– *How to determine the required SDCE tasks*

– *How to develop and refine the SDCE schedule and effort estimates*

– *How to work with the overall source selection structure and the SSEB*

– *How to assist with the development of evaluation standards*

– *How and when to prepare the SDCE input for the source selection plan and the source selection evaluation guide*

– *How to plan for the disposition of SDCE data, including that needed for metrics*

4.C.1 Outline SDCE Tasks and Schedule

4.C.1.1 Initial SDCE Planning and Scheduling

In the early phases of planning, it is likely that only the SDCE team leader will have been identified. Therefore, the first few steps in the planning activity are organized to reflect this limited level of staffing. The SDCE team leader should be involved in the preliminary source selection planning and must understand the overall source selection schedule and influence it as necessary to accommodate essential SDCE tasks. The specific relationships of the various SDCE activities and approximate timeframes for their completion are shown in figure 4-1 as a starting point for developing the actual, detailed schedule for a given source selection. Figure 4-1 presents two schedules, representing source selections of different durations and complexity. The guiding principle for this phase is that the source selection must be conducted as quickly and efficiently as possible, while ensuring a sound and equitable evaluation and selection process.

One of the essential considerations at this point is how to handle site visits as part of the SDCE. Several scenarios, recommendations, and cautions are contained in this section. One of the key decisions is whether the source selection is being planned for "discussions" or for "no discussions." This decision will generally be made by the Source Selection Authority (SSA) or SSEB chairman and will affect details of the SDCE planning. The planned site visits, conducted by a single team, must fit within the window for possible discussions between the initial competitive range determination and the request for best and final offers (BAFOs). There may be further constraints of multiple competitive range determinations and a cutoff date for CRs and DRs before the request for BAFOs. If a schedule problem exists, either the overall source selection schedule must be extended or the length of the SDCE site visits must be reduced. The SDCE team leader must work closely with the program office to estimate the number of bidders and the size, scope, and schedule of the SDCE effort required.

4.C.1.2 Source Selection Structure

The SDCE team leader must interact with the SSEB chairman and other SSEB members to understand and influence the technical structure of the source selection (Areas, Factors, and Subfactors) and the relative priorities, assessment criteria, and evaluation standards for these items. The SDCE model is structured so that it can be merged with the hierarchical structure of the source selection. Placing the SDCE at the next level under "systems engineering" or "technical" (or the equivalent) is the preferred approach for most acquisitions. Figures 4-2 through 4-4 show how the SDCE structure and the source selection structure can be integrated in several different ways. In a complex source selection, there may be as many as a dozen Factors and several dozen Subfactors. Additionally, some of the Subfactors may be subtiered into Elements. For such a complex source selection, the SDCE should be attached as one of the Factors. For this case, evaluation standards involving the SDCE could be written at the Functional Area level. In less complex source selections, it is still recommended to attach the SDCE as a Factor where possible, but it may be necessary in some cases to attach the SDCE as a Subfactor. This may also be true when the source selection is organized around an Integrated Product Development concept that shows management and technical entries as Factors. In this case, the SDCE could be attached as a Factor at the same level as technical and management, or attached as a Subfactor under technical. When the SDCE is a Subfactor, a single evaluation standard would typically be written up for the SDCE as a whole.

The decision on how to organize the SDCE items among the criteria of the source selection is a critical decision that must be made early and with some care. (See paragraph 4.C.3.1 for more information on evaluation standards.) The organization adopted will largely control the amount of visibility the SSA has into the various software capabilities and how the identified strengths and weaknesses may influence the outcome of the source selection. Development of the recommended source selection

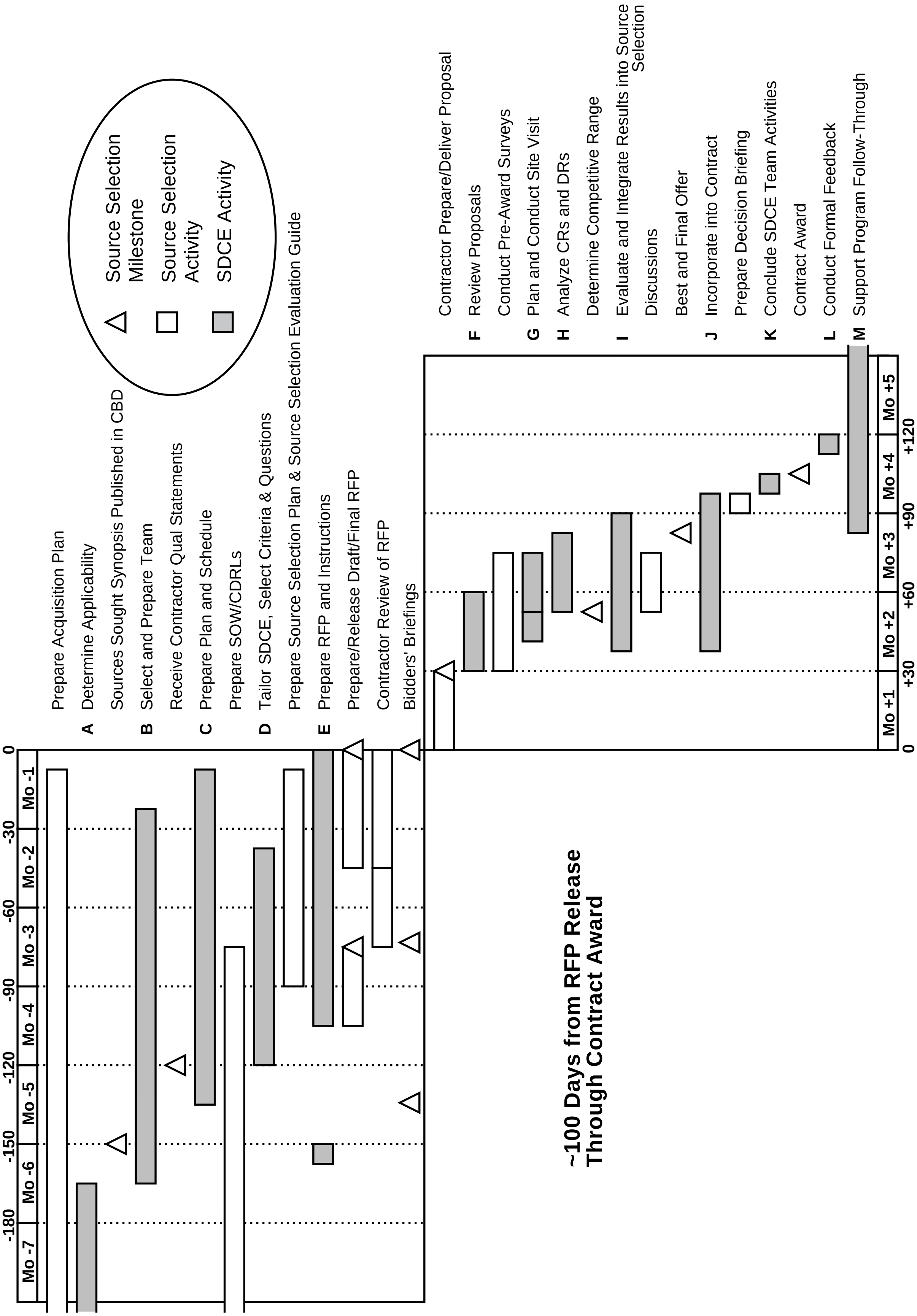


Figure 4-1. SDCE Schedule Integrated with Source Selection Key Events

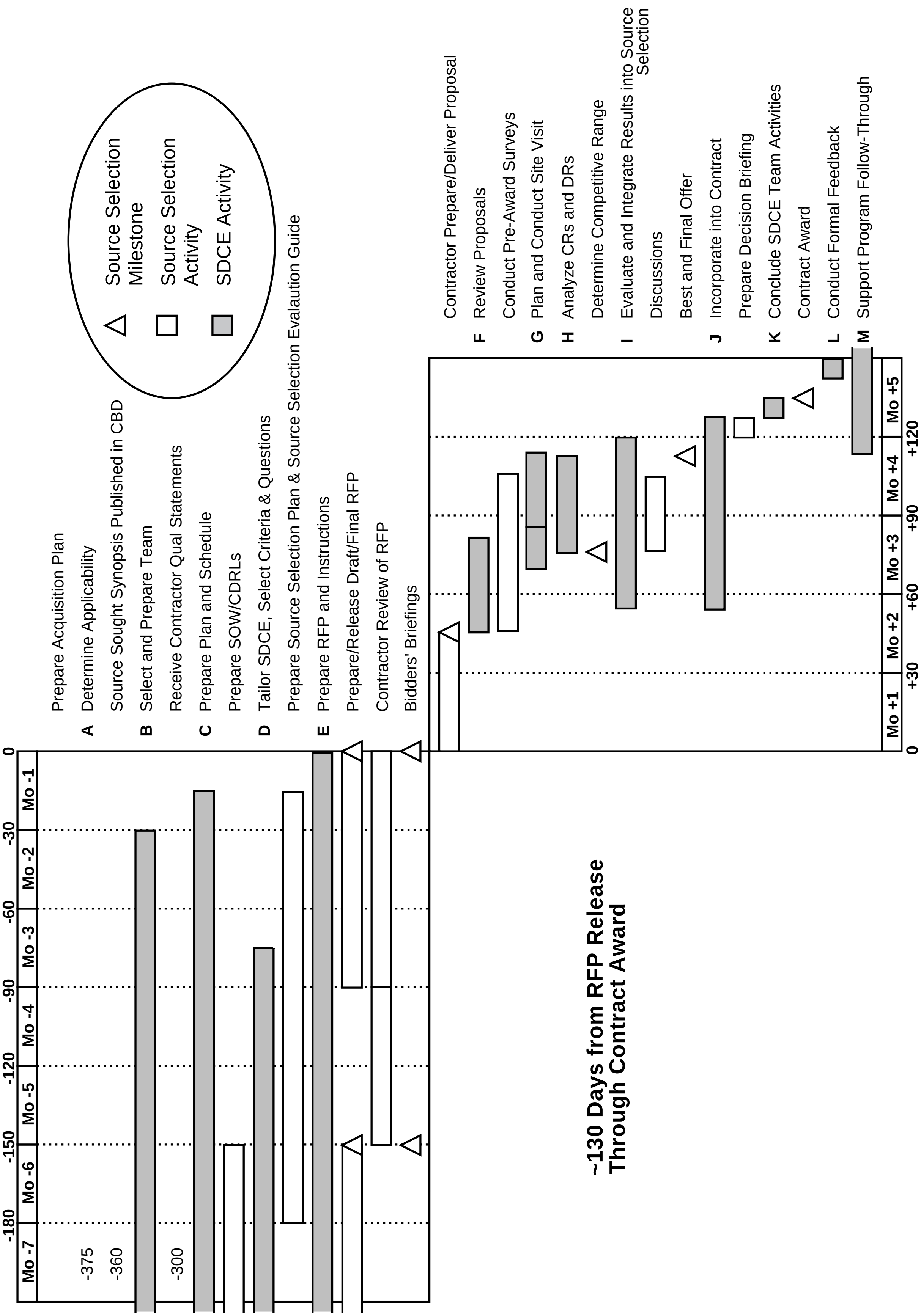


Figure 4-1. SDCE Schedule Integrated with Source Selection Key Events (Continued)

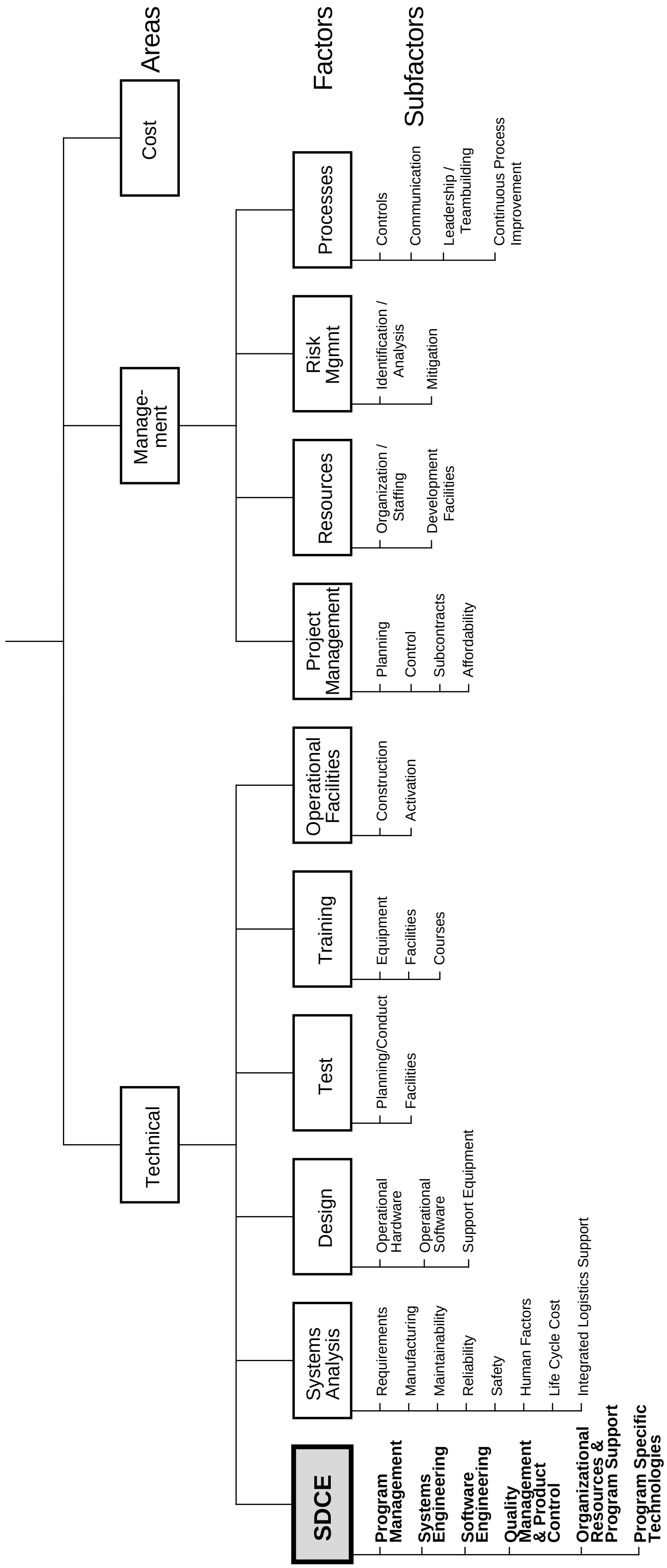


Figure 4-2. Placement of the SDCE within the Source Selection Structure – Large Project with 30 Subfactors

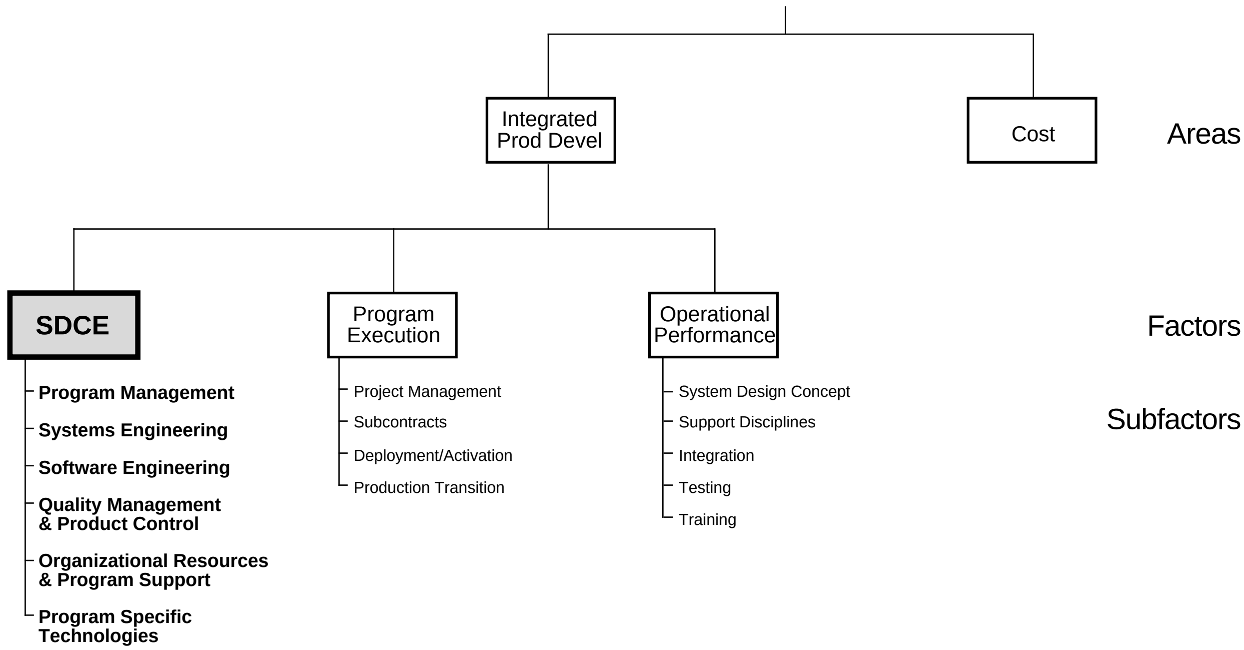


Figure 4-3. Placement of the SDCE within the Source Selection Structure – Medium Project with 9 Subfactors

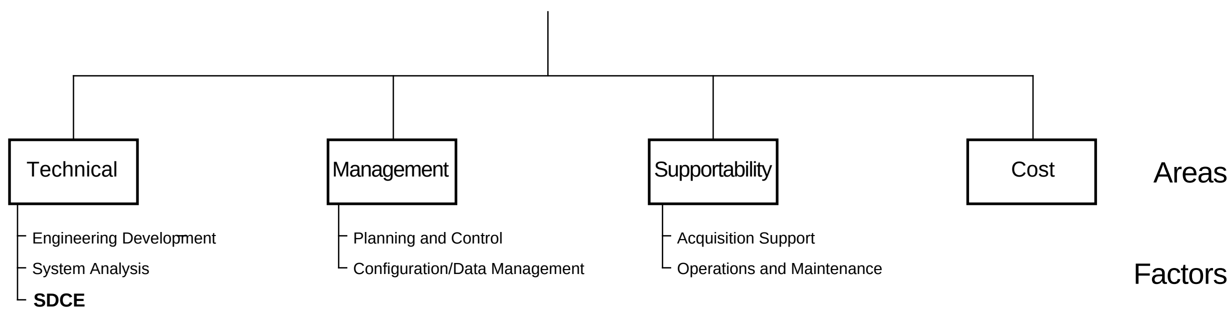
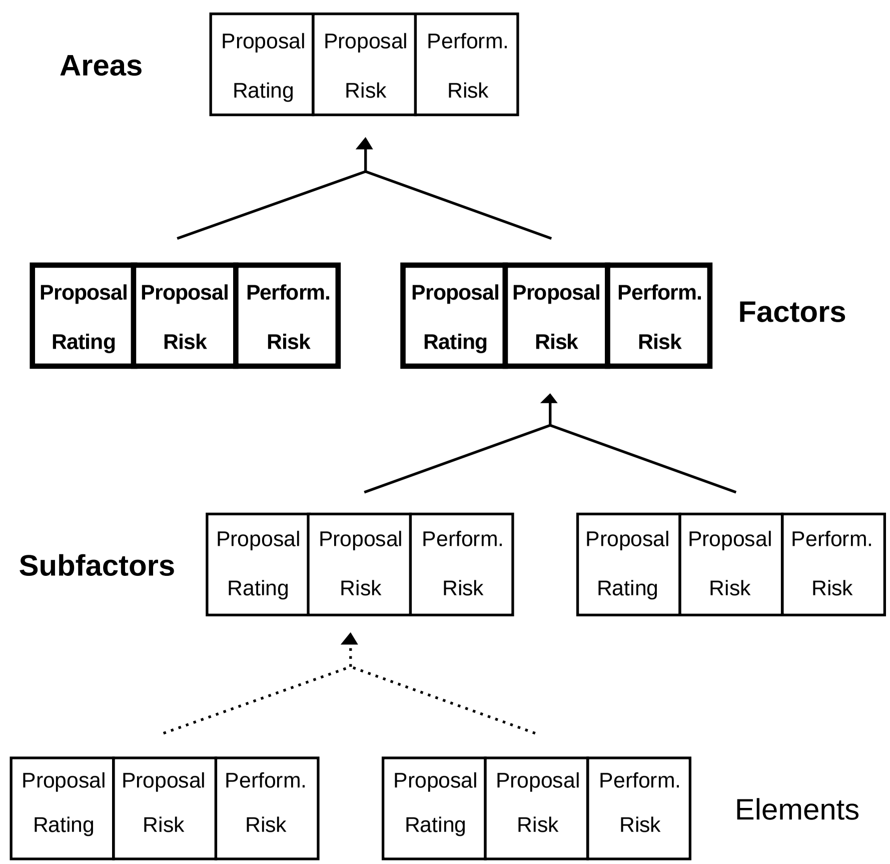


Figure 4-4. Placement of the SDCE within the Source Selection Structure – Small Project with 6 Factors

structure should also be made with awareness of how proposal ratings, proposal risk, and performance risk are combined. Figure 4-5 shows how this is typically done.

4.C.1.3 Source Selection Information

In the interest of economy and efficiency for both the government and the potential bidders, what is requested for the major proposal volumes and what is requested for the SDCE need to be coordinated to avoid duplication of effort and data. In particular, the government should not request the same information in multiple volumes, and the offeror should be encouraged to reference other proposal material where possible rather than to submit redundant material for the SDCE.



1. Proposal Rating is scored using: Blue, Green, Yellow, or Red.

2. Proposal and Performance Risk are scored using: High, Medium, or Low.

3. All three types of scores are supported by written narratives documenting Strengths and Weaknesses.

4. Proposal Rating, Proposal Risk, and Performance Risk are weighted equally.

5. Factors are the most fundamental layer:
   - Relate to important program characteristics.
   - Must be evaluated and scored.
   - Evaluation Standards are Usually written at this level,

6. Factors can be grouped into areas to facilitate evaluation, organization and presentation of results.
   - Areas are optional but typically employed.
   - Are used to group factors into very broad areas, typically:
     - Cost
     - Management
     - Technical
   - Must be formally rolled up from the Factors, if used.

7. Subfactors are optional but often employed:
   - Provide helpful suborganization when factors are too broad.
   - Evaluation Standards may be written at this level also.
   - May be evaluated or scored less formally than factors:
     - No specific Evaluation Standards.
     - May use other techniques than colors etc.
   - Subfactors can be further subtiered into Elements if necessary.

8. A total score is optional but almost never calculated, the SSAC and SSA make comparisons at the factor level and sometimes at the area level.

Figure 4-5. Summarization and Roll-up within the Source Selection Structure

4.C.1.4 Site Visit Scenarios

Although developed with a preferred baseline approach in mind, the SDCE can be applied in a number of ways. In particular, there are several options for conducting the site visit – often the most schedule-intensive part of the process. Additionally, some parts of the SDCE can be used outside the specific time window of the formal source selection. In those cases where it is not possible or not advisable to conduct a site visit during source selection, some of the benefit of face-to-face discussions can be obtained by conducting site visits in other contexts. This subsection identifies and defines some alternatives, lists conditions and assumptions associated with each alternative, and recommends when a particular alternative should be used. Additional alternatives and constraints may be applicable for a given source selection; the choice of approach for the procurement at hand needs to be carefully coordinated with the program office and the PCO. Each of these scenarios, except the sole source and Dem/Val scenarios, applies to competitive, major modifications to existing systems.

Preferred Baseline Approach – Preplanned Site Visit. This is the standard approach defined in the SDCE method. It is assumed that an open, competitive source selection is conducted, and that the acquisition source selection plan is not restricted with a requirement for "no discussions." The site visits are conducted by the SDCE team; this team is an integral part of the SSEB and reviews the proposal information prior to conducting the site visits. The site visits can be conducted as soon as the SDCE team has reviewed the proposals and the SSEB chairman makes a decision to open discussions with the offerors. It is also possible, based on the clarity and quality of proposals (including the SDCE proposal information), that a decision will be made that neither discussions nor site visits are necessary for the procurement at hand.

Source Selection Preplanned for "No Discussions." This is the same as the preferred baseline approach except that the RFP and the source selection planning are structured from the outset to permit award without discussions. For the SDCE, this primarily means that the amount of available time to conduct site visits may be limited to less than half of that available if the site visits could start soon after the receipt of the proposals. (Site visits are conducted only after a decision is made to open discussions; this decision is typically not made until the middle of the proposal evaluation phase.) Under this approach, the RFP states that a site visit may be conducted if the SSA determines that it is necessary to open discussions.

Sole Source New Development or Major Modification. Where contracts are awarded sole source, an evaluation is not required to discriminate among competing offerors, but the SDCE (including a site visit) is still beneficial to review the offeror's capability and to solicit a contractual commitment to follow proposed processes. Although detailed scores are not as important, early insight into risk areas and the ability to address, and possibly correct, weaknesses in the proposed approach could be very helpful.

Site Visits During the Demonstration/Validation Phase. If the EMD phase is preceded by a Dem/Val phase, part of the SDCE process can be applied in advance of the EMD source selection. Two useful approaches are described below. In either case, it must be remembered that the option must be preplanned and the necessary tasks need to be written into the Dem/Val statement of work to support the necessary SDCE activities and deliveries of data.

One or more dry run or practice SDCEs can be performed during the Dem/Val phase. Because of the possibility that a new contractor, who was not one of the Dem/Val contractors, might bid for the EMD contract, it is important not to constrain or unduly bias the source selection process. Therefore, the results of the Dem/Val dry runs cannot be integrated directly into the EMD source selection. Even so, the materials collected can be saved and filed by the contractor, and then the appropriate parts can be submitted with the proposal and used during the site visits associated with actual EMD SDCE and source selection. The major benefits of this technique include promoting early mutual understanding between the offerors and the government about important program capabilities; stabilizing and baselining processes well prior to actual need; and markedly increasing the efficiency of the upcoming EMD source selection, SDCE analyses, and site visits.

In the special case where the acquisition strategy is to downselect the EMD contractor from among the Dem/Val contractors, much of the SDCE process, including the site visit, can be accomplished with each contractor team as part of the Dem/Val phase activity. The information obtained from this type of SDCE process can be incorporated directly into the EMD source selection in the same way that technical data and evaluation results from “flyoffs” and competitive prototyping are handled. The exact details of this process vary from program to program and from acquisition center to acquisition center, so that the details need to be carefully coordinated with the local contracting authority and program office. This method is *highly recommended* when it is available. The major benefits of this technique include a significant reduction of effort during the critical EMD source selection period and the ability to work closely with the contractors for extended periods in a more benign environment. Additional benefits include promoting early mutual understanding between the offerors and the government about important program capabilities, and stabilizing and baselining processes well prior to actual need.

Post-Award Site Visits. Independent of whether a site visit is conducted prior to contract award, a post-award site visit could be conducted to review the contractor’s capability, planned processes, and resources to be applied on the program contract. In these cases, the objective of the site visit is not to reduce risk in selecting a capable contractor, but to reduce risk in executing the program within its baselines. To be most useful, since the degree of risk is not known prior to contract award, the program office should have the desire and ability to set aside management reserve funds to work with the contractor to correct significant weaknesses in capability discovered after contract award.

4.C.1.5 Wrap up Preplanning Activities

The preliminary information developed during this phase of planning will allow the team leader to estimate the schedule needed for the various activities and tasks of the SDCE and begin developing the required SDCE inputs to the source selection plan, at least at the table-of-contents level.

4.C.2 Prepare Preliminary Plan

4.C.2.1 SDCE Cost Estimate

Additional planning tasks can be undertaken as soon as the SDCE core team has been selected. It is important to provide to the SSEB chairman and the program office a fairly definitive estimate of the costs involved for the SDCE process. Based on the number of bidders and the scope and size of the program, manpower, travel, and support estimates can be made. The cost estimate can be based on the typical case in table 4-6.

Table 4-6. SDCE Cost Estimate

For a large-scale source selection taking 130 days from RFP release until contract award, the SDCE team might include one team leader and five additional members. Up until six months prior to RFP release, the only team member is likely to be the team leader or program office representative. The level of effort prior to this time is minimal and easily included within the variations of the overall source selection effort estimates. At about six months prior to RFP release, the SDCE tasks start to be a noticeable percentage of the total effort. Key tasks prior to RFP release include training/familiarization, tailoring, and RFP preparation. The high-expenditure tasks after RFP release include: preparing for proposal review (including further training/familiarization), proposal review, conduct of site visits, and preparation of results and findings. A recommended cost estimating methodology is to develop a staffing profile based on the tasks and schedule, and then to predict the travel cost based on estimated number of site visits, typical duration of site visits, and assumed locations of site visits. The tabulation below presents an estimate of the government effort required,organized by activity; the time-phased profile must be developed according to the schedule of each unique program

Activity	Effort (Person Days)
SDCE team preparation through final RFP release	20– 80
SDCE team proposal analysis prior to site visit (per offeror)	18– 24
SDCE team site visit (per offeror), plus travel/support cost	24– 30
SDCE team evaluation (per offeror) after site visit	18– 24

4.C.2.2 Integrated Planning

Integrating the SDCE planning with the overall source selection planning is essential. Specific data inputs and support of integration and review meetings will be required. The SDCE information should be incorporated directly into the source selection plan. If the SDCE team (or at least the leader) is on board early enough, this level of total integration can be achieved. An alternative is to prepare separate volumes or attachments to the source selection plan. Using this alternative requires extra care to ensure that the SDCE material and overall planning material are consistent and that the SDCE material is reviewed and controlled by the SSEB.

4.C.2.3 SDCE Data

Consideration must also be given to the large quantities of sensitive data involved with conducting an SDCE and the handling and ultimate disposition of this data. Key decisions include determining what data is needed, how many copies, who will get the data, who will be provided access, which data will be retained, which data will be returned to the contractor, and which data will be destroyed. Of particular importance is the determination of which data will be needed to support the preparation of meaningful metrics data, as outlined in section 4.K.

In addition, all source selection data or documents having a direct relationship to the source selection action must be protected and appropriately marked. The SDCE team does not generate many of these, but since they will handle these documents, care must be taken to plan for their use and disposition by the SDCE team. Table 4-7 itemizes the records that are particularly source selection sensitive. For these cases the authority to disclose source selection information is vested in the Deputy Assistant Secretary of the Air Force for Contracting, and the responsible Program Executive Officer, Designated Acquisition Commander, or the Activity Commander for the specific contract or records involved.

Table 4-7. Types of Source Selection Sensitive Data

Source list screening criteria
Results of screening, justifications for any non-solicitation
Source selection plan
* Weights and standards
* Narrative assessments
Documentation of SSAC and SSEB members
* Proposals, amendments, and alterations thereto
* Summaries of oral presentations made to the SSEB
* Evaluation reports, including independent assessments
* Inquiries sent to offerors by the SSEB and responses thereto
* Deficiency reports, clarification requests, and responses
* SSEB evaluation report
* SSAC analysis report
* Company-specific past performance information
* Source selection presentations (viewgraphs and text)
Records of attendance at decision briefings

* Normally require continued protection even after contract award

4.C.2.4 Contractor Organization and Teaming Alternatives

One of the essential considerations at this point is the anticipated teaming arrangements of the various bidders. The SDCE method is designed to handle a wide range of organizational possibilities that offerors may propose. Current practice on embedded software systems ranges from the simplest cases where a single, localized organization develops all the software to the most complex cases where multiple contractors, in some interrelationship, participate in the development and integration of the software. This subsection delineates representative cases and recommends how the SDCE might be applied to each one. It should also be noted that, for a given procurement, different offerors may propose different organizational arrangements. Thus, the techniques outlined below may have to be tailored and combined for application to a given source selection. For all cases, several key groundrules should be observed:

- All major software developers should be evaluated.
- Questions about unique or different processes should be answered individually by the participants.
- Questions about common processes can be answered once, covering the joint relationship(s).
- Even when an offeror proposes common processes, evidence should be provided individually.

Single Organization Bidder. In this scenario, all the software is developed and integrated by a single contractor, within a single organization, and at a single site. The SDCE is applied to this single organization.

Multiple Organizations as a Single Team. In this scenario, the software is developed and integrated by multiple contractors, within multiple organizations, and possibly at multiple sites. The various parties are highly merged as a team, and the contractors, organizations, and sites are all known at the time of the source selection. In this case, the SDCE is applied to the whole team. The focus must be on how the team, as a cohesive unit, plans to do business, rather than on the specific individual capabilities of the various team participants. Typically, a single set of data is collected and the site visit is at a single location chosen by the contractor team. If the particular teaming arrangement is new, there may be no historical data on how well the combined team capabilities work. Therefore, evidence must be collected from the various team participants, and the evaluation by the SDCE team will require considerable engineering judgement. As in all cases where detailed, applicable evidence is not available, the offerors must explain why their selected approach was chosen from among the alternatives.

Single Integrator and Developer with Suppliers or Vendors. In this scenario, the major items of software are developed and integrated by a single contractor or team, but specific, relatively minor or localized items may be acquired from suppliers or vendors. The developing and integrating contractor or team is known at the time of the source selection, but some suppliers and typically all of the vendors are selected by the lead team at some later time. For this case, the SDCE should be applied to the contractor or team as it is known at the time of the source selection. In addition, special emphasis must be placed on evaluating how the suppliers or vendors will be selected and on how their processes and products will be integrated into the mainline effort. For example, the lead contractor team might use the technical information of the SDCE method (without the government source selection-specific items) to conduct preselection evaluations of its suppliers. Alternatively, the items acquired may be purchased essentially “off the shelf,” and detailed evaluation of the vendor’s development capability would not be cost effective.

Prime/Integrator with Multiple Subcontractors. In this scenario, the prime contractor performs the integration function, and possibly some of the development, but major portions of the software are developed by subcontractors. Some of the subcontractors may be known at the time of the source selection, and some may be scheduled for later selection. The known subcontractors may be organized to work closely with the prime contractor as team participants or may plan to work somewhat independently. Some aspects of this case are analogous to the other cases described above and should be dealt with as outlined in those descriptions. However, there is a new possibility in this arrangement not covered by the previous descriptions: the known subcontractors may not be highly merged into the lead team. In this case, it is recommended that separate sets of data, focused on the technical and managerial content of their assigned portion of the whole, be obtained from each of the team participants. Separate site visits are also recommended for each site or organization involved. Arrangements for performing site visits with subcontractors must be made through the prime contractor. The prime contractor is legally entitled to be involved and must be invited to the site visit and allowed to participate in the interaction with the subcontractor. However, the prime contractor representative is not a member of the SDCE team and cannot be allowed to participate in the preparation of results or in making judgements relative to the source selection.

For teaming arrangements in which subcontractors will be developing relatively minor or localized portions of the software, it may be appropriate for the prime contractor to invite the subcontractors to participate in the site visit to the prime, in lieu of site visits to each of the subcontractors. In this case, the prime contractor may wish to organize the agenda in such a way that the subcontractors can participate in the portions that are relevant to them and then be excused from proceedings that do not involve them or that may cover items that are considered proprietary to the prime contractor.

4.C.2.5 Complete Initial Planning and Scheduling

As the overall source selection planning is maturing and as the specific tailoring and scoping tasks for the SDCE are being accomplished, the SDCE team will be in a position to develop a definitive SDCE schedule and a tentative site visit agenda and schedule.

A final step in this phase is to provide detailed inputs to the source selection plan or to prepare a separate SDCE implementation plan. Volume 2, attachment 2-1 contains content recommendations and an example plan.

4.C.3 Update and Integrate Plan

After the SDCE tailoring activity is completed and all the team members have been selected, the SDCE planning activity can be completed and the SDCE source selection evaluation guide inputs can be prepared. The last steps of the planning include updating the plan with definitive scheduling and tailoring information, cleaning up the final inputs to the overall source selection plan, and reviewing the document with the source selection team.

4.C.3.1 Preparation of Evaluation Standards

By regulation, all proposals must be evaluated against standards rather than each other. It is therefore essential that these standards be clearly identified and documented prior to release of the RFP. Detailed evaluation standards must be prepared for all items at a designated level of the source selection structure (typically the Factor level). When these involve items from the SDCE model (typically FAs), the evaluation standards must be prepared or augmented by the SDCE team. The key attributes for these evaluation standards are that they must relate to the model criteria and to the program at hand. Depending on the program requirements, they may be direct quotes from the model criteria or may be tailored, augmented, or expanded for the source selection at hand. These standards are one of the most important parts of the source selection and must be carefully prepared with that thought in mind. Volume 2, attachment 2-2 contains several examples showing how evaluation standards can be associated with the various SDCE model elements.

4.C.3.2 Source Selection Evaluation Guide

Although a particular format for evaluation standards is not specified by source selection guidance, each acquisition agency has, as part of its culture or local process, something like a source selection evaluation guide (SSEG). In practice, all the evaluation standards, including the SDCE-related standards, are documented in this guide. This document is an expansion of the source selection plan and is prepared by the core SSEB and program office personnel for use by the selection evaluators. The main focus of the SSEG, above and beyond the source selection plan, is to provide additional detailed procedural data for conducting the evaluations, to provide essential administrative information, and to document the evaluation standards. The SDCE evaluation standards must be merged into the source selection evaluation guide along with the other evaluation standards. Table 4-8 contains an outline of a typical SSEG.

Table 4-8. Source Selection Evaluation Guide Outline

ORGANIZATION OF THE SSEB
Organization chart, duties, and responsibilities and limitations of each member or function of the SSEB.

SECURITY
Identification of the office security manager and accountable officer, procedures, and general instructions for accountability and destruction.

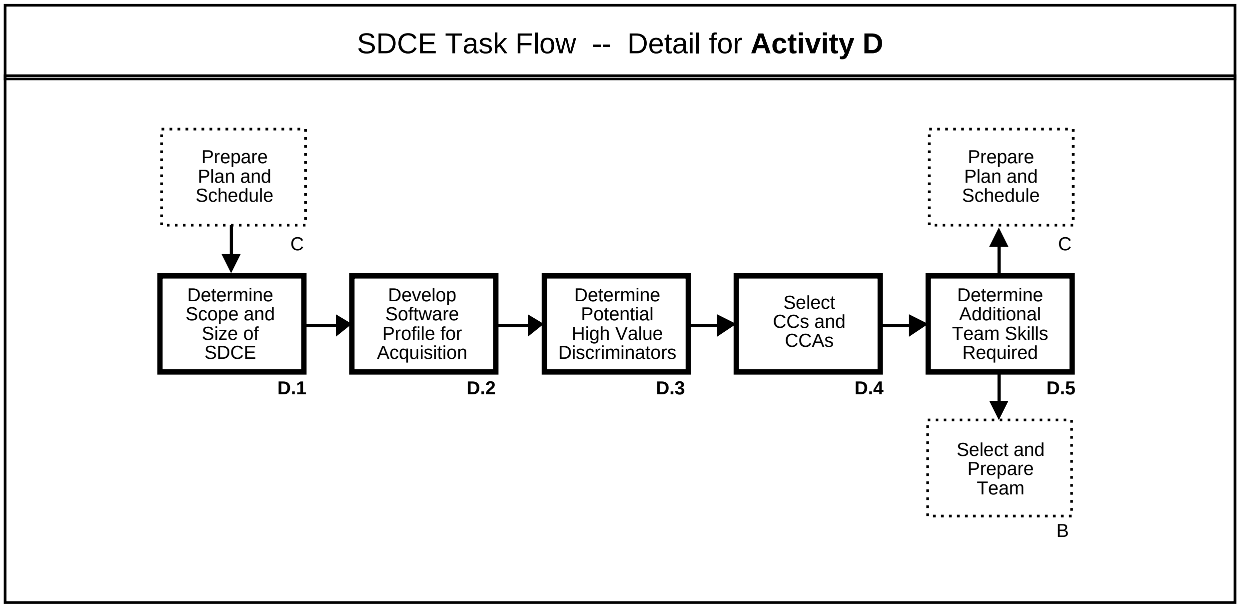
ADMINISTRATION
Description and location of area, hours of operation, how proposals will be controlled, how paper will flow within the SSEB, who will control access to the area, how liaison will be managed between the Cost Panel and the Evaluation Panel, and a detailed (daily and hourly) schedule of evaluation.

EVALUATION PROCEDURES
Complete descriptions of how the Evaluation Panel will function from receipt of proposals through conduct of debriefings, how cost will be evaluated, and how past performance will be evaluated. Identification of which source selection forms will be utilized, and how evaluators will complete these forms. Definitions of color codes, strengths, weaknesses, deficiencies, and clarifications. Requirements for evaluation of manpower and determination of risk assessment and most probable cost.

BRIEFINGS/REPORTS/DECISION DOCUMENT
Description of contents of briefings and steps necessary to be prepared to conduct briefings, identification of those individuals (or functions) who may attend briefings, requirements for review of materials, identification of report requirements and content, and description of the decision document.

STANDARDS
Complete evaluation standards for all specific criteria as well as standards for past performance as applicable.

Section 4.D Tailor SDCE, Select Criteria and Questions



This section describes the approach for determining the scope and the appropriate elements of the SDCE model to be used for a particular acquisition. The tasks described will assist the SDCE team in performing the following tailoring functions:

- *How to develop the software profile for the acquisition at hand*
- *How to determine the high value discriminators for the acquisition*
- *How to select the criteria and questions that qualify the discriminators*
- *How to identify other skills required by the team*

4.D.1 Determine Scope and Size of SDCE

To be generally useful, the SDCE model covers all capabilities and domains. In applying the SDCE to a particular program, it is essential to tailor the model to that program. From the acquisition point of view, all areas of the SDCE model are not high value discriminators on all programs. Furthermore, effective use of the available time during a source selection requires careful tailoring to focus on the key capabilities germain to the program at hand. From the offeror's point of view, it is critical to focus proposal resources on process descriptions and commitments essential to program success. The SDCE team must, therefore, identify the high value discriminators and the primary areas of potential risk, and tailor the SDCE accordingly.

There are numerous factors that need to be considered in tailoring the application of the SDCE to a particular acquisition. Program characteristics vary widely, and the SDCE should be tailored to evaluate the high value discriminators, thereby reducing the risk inherent in the source selection process. In applying the SDCE method, program offices should request support from their Center

SDCE OPR. This support should include help in tailoring both the model and the application of the SDCE to the specific program. The steps contained in this section will help guide the team through this process and provide other information to facilitate the team's thinking during the tailoring activity.

The scope of an SDCE is determined by four factors: (1) the number of CCs determined to be applicable, (2) the number of contractors to be evaluated, (3) the number of subcontractors to be evaluated, and (4) whether a site visit will be conducted.

4.D.2 Develop Software Profile for this Acquisition

A system/software characteristics analysis will establish the basis for determining the elements of the SDCE model (via the tailoring activity) for a particular program. It will also provide a basis for comparing the offerors' interpretations of the program requirements with that of the program office.

The first step in tailoring is to determine and describe the characteristics of the software development for the specific program. The SDCE team, in conjunction with the program office, should identify the characteristics listed below. This information will help determine whether an offeror's projects submitted as examples of processes are comparable to the proposed project. The Cover Sheet for Project Sample Data (Volume 2, attachment 3-3) can be used as a guide for developing this profile.

- Estimated software size in terms of the developed and delivered lines of code. Use of COTS, reuse, and other factors and assumptions should be reflected in this estimate range.

- Program development schedule from contract award to delivery of an initial operational capability. This would be typically defined in the RFP. Also, the range of software development schedules could be estimated. This would be the time from a complete software specification review through completion of the last computer software configuration item (CSCI) integration and test.

- Rough order estimate of the anticipated software development team size, using recognized software estimating models.

- Estimated software complexity and relative development risk based on an awareness of the extent to which the particular system and software development requirements are unprecedented.

The second step in the tailoring activity is the identification of special technical requirements and factors, to include, but not limited to the following.

- Software development language requirements and associated tools and methods
- Systems/software engineering environment
- Complex integrated circuit development (VHSIC/VHDL)

- Open systems architecture
- Commercial off-the-shelf software
- Reuse requirements
- Complex interface requirements
- Security/safety requirements
- Portability

4.D.3 Determine Potential High Value Discriminators

A key element of tailoring is determining the high value system and software development capabilities (discriminators) as a function of the specific acquisition program. This subsection identifies a generic set of these high value capabilities, but there may be other major program-peculiar discriminators which should be considered in the tailoring effort.

Phase of the Program. The SDCE needs to be tailored as a function of the phase of the program and the extent of the software development, since it is designed to support source selections involving software development for mission critical computer resources. Software for a new program is most commonly developed in the EMD phase, and it is this phase which the SDCE is primarily designed to support. However, if significant software development is planned as part of a production or major modification program, it would be appropriate to conduct an SDCE even if the effort is a follow-on, sole-source contract modification. In this case, coverage of existing stable processes in use and understood by the acquisition organization could be tailored out.

RFP Process Requirements. The SDCE is intended to review the offerors' capability to do systems and software development in terms of processes, practices, methods, and tools. The program RFP will typically include specific engineering process requirements. These specific requirements must be carefully reviewed to tailor the SDCE model to be consistent with the RFP requirements. Examples of these requirements include the following:

- Integrated product development teams
- Incremental software development
- Software modularity and reuse requirements
- Systems engineering master schedule
- Tailored data item descriptions
- Joint Air Force, Army, Navy programs

Domain Application Requirements. Although the SDCE is designed to support AFMC MCCR system acquisitions in general, there are significant differences in application domains between MCCR systems. The SDCE must be tailored to reflect the most critical capability requirements for the program at hand, considering the application domain. Each of these domains includes unique requirements and processes that must be considered in tailoring the SDCE to achieve the maximum benefit to the acquisition process. Table 4-9 is a partial list of these domains and their characteristics and requirements. The domain characteristics are typical and are provided to assist the SDCE team in determining the high value discriminators and subsequent CCs. The list highlights certain domain characteristics that might be helpful to consider in tailoring the SDCE to a particular program. It is not complete, nor is it intended to identify characteristics that are exclusive to a domain.

Security Requirements. Many programs have strict security requirements that have a major impact on systems engineering and software engineering capability and process. The SDCE should be tailored to consider important security requirements. These might include the following:

- Trusted systems software
- Multilevel security software
- Configuration and architectures to facilitate operational and support considerations with classified software
- Managing and maintaining systems secure from intrusions

Nuclear Surety and Safety Issues. Many development programs have critical human safety requirements which impact software design capability and, in particular, design, implementation, and verification processes. The SDCE should be tailored to consider these concerns as a function of the requirements of the program at hand. For example, capability questions should be included to address systems and related software engineering requirements, and design, implementation, and verification processes to assure safe system operation.

Once the high value discriminators are determined, the SDCE team should perform a review to verify that the SDCE has been properly focused on the software engineering and related systems engineering capabilities that are important to the program at hand.

4.D.4 Select CCs and CCAs

The next step in tailoring the SDCE is to select CCs based on the program profile and high value discriminators identified in subsections 4.D.2 and 4.D.3. Because of the model structure, selecting specific CCs implies the selection of the encompassing CCAs and FAs as well.

A candidate list should be developed by listing all those CCs that are appropriate for the acquisition at hand. The list may be added to, or deleted from, as the SDCE team determines. The team must use its experience and understanding of the software elements of the system being acquired to finalize the list to suit the particular acquisition. Points to keep in mind:

Table 4-9. Some Domain Characteristics

1. Avionics
1. Real-time man-in-the-loop interactions, real-time executives
2. Safety of flight, e.g., terrain following radars
3. Space- and weight-limited computer throughput and memory requirements
4. Highly concurrent hardware (non-computing) and software development
5. Complex distributed systems architectures
6. Closely coupled with systems engineering process
7. Solve very fast complex signal processing problems
8. Security

2. Flight and Engine Controls (Air Vehicle Management)
1. Redundant computer architecture and software design requirements
2. Safety of flight, critical requirements
3. Closely coupled with systems engineering process
4. Critical verification requirements

3. Simulators (Trainers), Crew and Maintenance
1. Ground based systems
2. Use of general purpose COTS computers
3. Use of COTS software
4. Real-time or close to real-time computations
5. Modeling and simulation computations
6. Large databases
7. Closely coupled with systems engineering process

4. Weapons Control Systems and Electronic Countermeasures
1. Nuclear certification
2. Safety critical
3. Verification/validation sensitive
4. Closely coupled with systems engineering process
5. Volatile requirements (threats)
6. Rapid reprogrammability

5. Command, Control, Communication, and Intelligence
1. Trusted systems software/multilevel security
2. Predominantly ground based
3. Large databases
4. Man-machine interactive real-time, near real-time

5. Command, Control, Communication, and Intelligence (Continued)
5. Signal processing oriented
6. COTS general purpose computers
7. Security
8. Safety
9. Fail safe
10. Real-time executive

6. Armament
1. Small, space-limited packaging
2. Nuclear certification
3. Safety critical
4. Large production volume, driven to conserve computing resources.
5. Multiple platform requirements

7. Space and Missiles
1. Redundant, fail safe, must-work-first-time designs
2. Control systems, safety critical
3. Verification/validation sensitive
4. Real-time executives
5. Airborne and space-borne computer qualified designs and components
6. Closely coupled with systems engineering process

8. Ground Based Support Systems
1. Use of general purpose COTS computers
2. Use of general purpose COTS software
3. Large databases
4. Non-real time

9. Ground and Automatic Test Equipment
1. Test requirement documents
2. Special Computer Programming languages
3. Test program sets
4. Units under test
5. Standardization and modularity
6. Use of COTS general purpose computers
7. Very large computer programs (KSLOC)

10. Codes and Cryptography Processing and Control
1. Security
2. Critical dictated solutions and algorithms
3. Critical verification requirements
4. Large databases
5. Trusted systems software
6. Redundancy, segregated control
7. Rigorous traceability and audit trails

- All CCs presented in the RFP should be evaluated during the proposal evaluation cycle prior to the site visit.
- All required SDCE information must be requested in the proposal. A site visit can only provide clarification and substantiation of information. It cannot be used as a vehicle to explore FAs/CCAs/CCs not addressed in the RFP and proposals.
- Reviewing process definitions against evaluation standards and criteria, and reviewing documentation that substantiates the use of the processes, are time consuming activities.
- The CCs reviewed during the site visit can be a smaller subset of the original list contained in the RFP. The CCs to be reviewed during the site visit are determined after proposal evaluation.

If a high value discriminator is not sufficiently addressed by CCAs and CCs in the model, the acquisition agency can develop program-specific CCAs and CCs to address the issue. New CCAs and CCs should be developed in accordance with the structure of FAs/CCAs/CCs in the SDCE model. Model criteria and a set of questions must also be developed.

This is a significant undertaking, and enough time and expertise should be devoted to this step to ensure a quality product. Before developing CCAs or CCs for an acquisition, the program office should contact the AFMC SDCE OPR to determine if CCAs or CCs similar to those needed have already been developed by some other organization. If not, the program office should involve the Center SDCE OPR and other functional expertise available locally in developing the required CCAs and CCs.

The new CCAs, CCs, criteria, and questions should be forwarded to the AFMC SDCE OPR for lessons learned, use by other organizations, and possible insertion into the model for future use. This is further discussed in subsection 4.K.2.

Having created a final set of CCs, the team should review the criteria and questions (chapter 5) associated with the selected CC to determine their applicability. Existing criteria or questions should be excluded only if they are not applicable to the acquisition at hand. To help ensure consistency of the SDCE method, criteria and questions should not be changed; required modifications should be handled as deletions and additions of new criteria or questions.

In addition to the CCs, criteria, and questions, the SDCE team needs to tailor the list of sample documents that would represent evidence of process policies, standards, training, and use. Table 4-10 lists the types of evidence generally suitable for various CCAs and CCs. The number of pieces of evidence needed to show that processes are actually being used as defined in an offeror's process definitions depends on criticality and perceived risk. For very large, complex, highly unprecedented systems (i.e., high risk), it would be appropriate to solicit up to three program examples to validate capability and process compliance. If, on the other hand, the program is relatively small, simple, and precedented (i.e., low risk), one sample of evidence showing that a process is being followed for each CC would be sufficient. For teaming or subcontractor arrangements, samples should be provided for each major software developer on the team.

Table 4-10. Tailoring Guidance for Selection of Samples of Evidence

Project Sample Data Criteria	CCAs and CCs
Software development plans	All, 3.1.4
Program organizational charts	1.1.1 & 1.1.2
Organizational charts from program manager through working-level software engineers	1.1.1 & 1.1.2
Contract work breakdown structures covering software development	1.2.2 & 3.1.2
Software work packages	1.2.3
Cost/schedule control system criteria report applied to software OR	1.2.3 & 1.2.4
Cost performance report applied to software	1.2.3
Systems through detailed software development schedule	1.2.3 & 1.2.4
Systems engineering master schedule showing software events and completion criteria	1.2.4
Subcontractor RFPs and SOWs defining software tasks	1.3
Maintenance contracts for providing software rights for post deployment	1.4
Risk management plans covering software	1.5 & 3.2
Systems and subsystem specifications	2.1 & 2.1.2
Tradeoff study reports addressing software	2.1.5 & 2.5.2
Requirements traceability matrices/tables	2.1.4 & 2.1.5
Design review minutes	2.2 & 2.3
Interface control specifications/documents	2.1.1
Systems engineering master plan	2.5.3
Systems engineering staffing plan/final report	2.5.4
Test and integration plans	2.6.1 & 2.6.2
Reuse plan covering software	2.7
Reuse tradeoff reports	2.7
Software size, effort, schedule, and cost estimates	3.1.1
Past actual productivity rates	3.1.1
Software tree structure CSCIs through units	4.7
Software status reports	3.2.2
Software requirements specifications	3.3.1
Software development folders/files	3.4.1 & 4.7
Peer review minutes/reports	3.5.1 & 4.5.2
Software integration and test plans	3.6
Software test procedure	3.6.2
Software quality assurance plan	4.1.1
Software discrepancy reports	4.1.3
Defect prevention plan	4.3.1
Software development metrics	4.4
Peer review plans	4.5.1
Internal independent verification and validation plans	4.6.1
Software configuration management plans	4.7.1
Software development, integration, and test facilities plans	5.2
Software training plans	5.3.1
Software staffing plans, including actual staffing profiles on completed programs	5.4.1
Software process improvement plans	5.6.1

This tailored list should be used as a guide by the offerors. However, the final determination of number of pieces of evidence, which team members will supply the evidence, and what constitutes evidence of use, is left to the offerors.

This effort results in an integrated package, which is included in the RFP for offeror response in the proposal, and in the source selection plan as appropriate. The package includes:

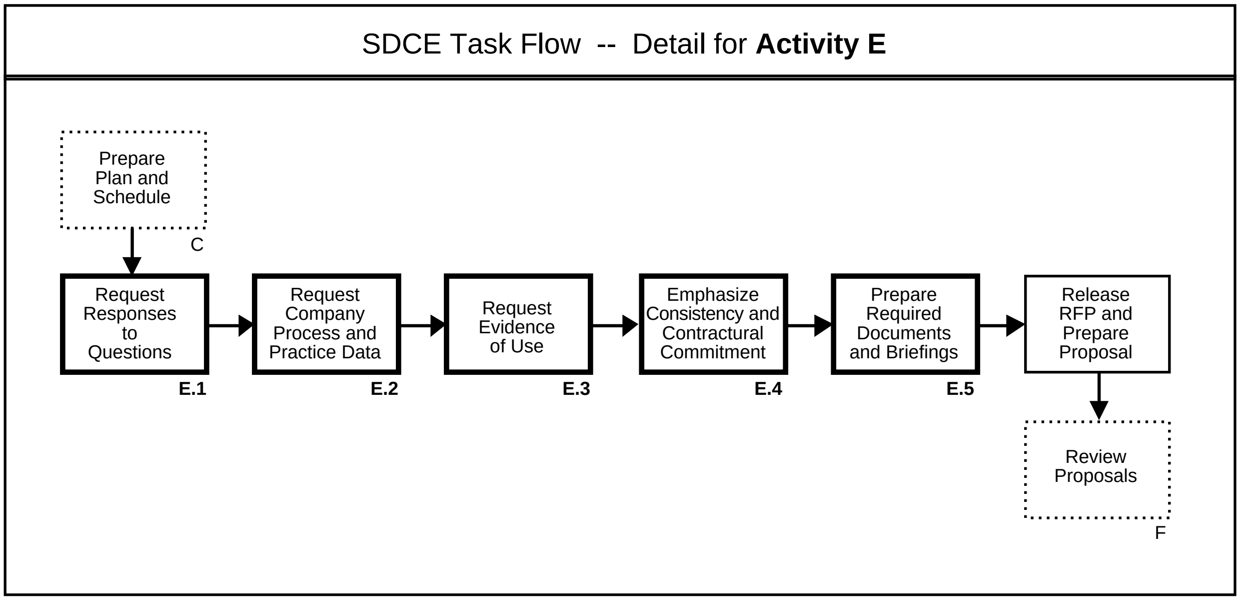
- Final list of CCs, including additions if appropriate.
- Questions for each CC, including additions if appropriate.
- List of suggested documentation requirements substantiating process use for each CC, including additions if appropriate and number of pieces of evidence requested.

4.D.5 Determine Additional SDCE Team Skills Required

The team leader is responsible for ensuring that a fully staffed and qualified team is used on the SDCE. This is the final step in building and training the team.

- Review the CCs on the final list and compare them with the experience of the current team members.
- Identify any shortfalls in the experience required.
- Acquire additional team members as necessary to ensure team experience addresses all CCs to be evaluated during the acquisition.
- Ensure that all members, including new members, receive the SDCE training if required (refer to section 4.B).

Section 4.E Prepare RFP and Instructions



This section deals with the incorporation of SDCE relevant information into the RFP. The topics discussed include:

- *How to request responses to questions, company practice data, and evidence of use*
- *How to specify required levels of consistency and contractual commitment*
- *How to integrate SDCE material into the SOW, CDRLs, and RFP*

4.E.1 Request Responses to Questions

The questions furnished by the government with the RFP are to be answered by each offeror following the instructions provided. Questions should be clearly grouped and identified by FA, CCA, and CC. Both hard copy and electronic media copies of the questions should be provided to the offerors.

4.E.2 Request Company Process and Practice Data

The offerors should provide, with their proposals, copies of the process, practice, and procedure manuals and internal standards that they intend to apply on the proposed project. Wherever possible these processes, practices, and procedures should comply with the SDCE model and should have been used successfully in the recent past on projects that have characteristics similar to the one being proposed. Volume 2, attachment 3-1, Capability Definition Matrix, should be included in the RFP package as a means of gathering this information.

New processes (that have not been previously employed by the bidder) should be described as well as the rationale for selecting them and the benefits and risks in using them on the program at hand.

4.E.3 Request Evidence of Use

The SPO should request supporting documentary evidence for responses which indicate the existence of an internal process, method, tool, etc. For instance, if it is claimed in an answer that the bidder has an institutionalized process for software size estimation, the document that describes this process and the order or instruction that implements that process should be provided. If weekly meetings of a configuration control board are claimed, minutes of a few successive meetings might be submitted. (See table 4-10 for a list of types of acceptable evidence.)

Evidence of use (samples) should be requested for the proposed processes, practices, and procedures for up to three prior or ongoing projects that have similar characteristics to the project at hand. The offeror should summarize this evidence in a completed Capability Implementation Matrix, using the template provided in the RFP package (Volume 2, attachment 3-2). The samples may consist of tailored instructions, metrics, minutes of meetings, evaluation reports, etc.

For each project for which a bidder provides evidence of use of processes, practices, and procedures, the SPO should request a Cover Sheet for Project Sample Data, using the template (Volume 2, attachment 3-3) provided in the RFP package. The cover sheet gives the government the information necessary to assess the applicability of the project used for evidence with the one being proposed.

4.E.4 Emphasize Consistency and Contractual Commitment

A key part of the SDCE method is to obtain contractual commitment for the capabilities proposed. The SEMP, SEMS, and SDP are the primary vehicles for securing that commitment. It is critical to emphasize that the proposed capabilities and the SDCE responses provide a consistent picture. Furthermore, the offerors should be encouraged to incorporate responses to the SDCE questions as much as possible directly into the SDP, SEMP, and SEMS. Responses to these questions would simply reference these documents.

4.E.4.1 Systems Engineering Management Plan and Master Schedule

The SEMP should describe the relationship of system and software engineering as proposed for the program at hand. The offeror should describe, in the SEMP, how the system and software processes are integrated and how the two functions interact in the development of a system and its software.

The SEMS should clearly indicate the temporal interrelationships (both sequential and parallel) of system and software functions and should show that adequate time has been allocated for all required processes. Completion criteria for the events should include system and software engineering process step completions.

4.E.4.2 Software Development Plan

The SPO should request each bidder to describe, in its SDP, the practices and procedures that it plans to employ on the proposed program. These practices and procedures should be consistent with the responses to the questions. If these practices and procedures need to be tailored for the program at hand, the proposed tailoring should be described, as well as any risk reduction activities to be employed to ensure success with the tailored process. (If the SDP data item description does not contain all the information required for an evaluation of the offeror's software process, then it should be modified and the modified version distributed with the RFP.) If necessary, SDP appendixes may be used to provide this information.

4.E.5 Prepare Required Documents and Briefings

As a part of the planning tasks, and as a result of the above considerations, several specific documents and briefings must be prepared. This subsection provides details and guidelines and references templates that may be used to facilitate preparation.

4.E.5.1 Prepare SDCE-Specific Input to Commerce Business Daily (CBD) Announcement

The purpose of including SDCE material in the overall program CBD announcement is to inform bidders that a SDCE may be conducted in the course of the source selection and that, if conducted, it will be an important factor in the source selection decision. If the decision to allow discussion and also to conduct an SDCE site visit is made prior to the release of the CBD announcement, the announcement can state that an SDCE site visit will be conducted. If, however, that decision is deferred, the CBD announcement should state that an SDCE site visit may be conducted.

Volume 2, attachment 2-3 contains wording that may be used or appropriately adapted and augmented in the announcement.

4.E.5.2 Prepare SDCE-Specific Input to Bidders' Briefing

The content of the SDCE part of the bidders' briefing should be consistent with the content of the source selection plan. As a minimum, it should include the topics described below:

Use of SDCE. This discussion may be similar to the CBD announcement but should stress the importance of the offeror's software development process and related system engineering management capability in the source selection. It should explain how the SDCE will be used in the source selection and how its results will be integrated into the source selection decision.

Overview of Model. The discussion of the SDCE Model should cover the Functional Areas and Critical Capability Areas sufficiently so that the audience will understand the model's conceptual hierarchical structure and its application to the evaluation. A copy of the model should be provided to all offerors. The briefing should not cover the model in all its detail but should point out the CCAs that will be important in this source selection.

Conduct of SDCE. Although the exact data to be gathered in a site visit will not be known until after a review of the proposals and associated documentation, the briefing should describe the procedure for scheduling site visits, the agenda for the site visits, how the site visits will be conducted, which of the offerors' staff (by function) should be available, and what kind of documentation might be requested by the evaluators. In general, the briefing should cover what the evaluators will expect from the offerors and what the offerors may expect from the evaluators.

Incentive Award Program. If an incentive award program is to be used on the contract and if that program is linked to the results of the evaluations of the offerors' software processes, then this program should be explained at the briefing. The award fees, schedule, method of evaluation, criteria to be applied, and time and amount of payment should be covered.

Required Documentation. The briefing should describe the software and systems engineering process documentation that will be required as part of the proposal and that will be analyzed by the government; the criteria that will be applied to the evaluation of the documentation; and the importance of evidence of use, institutionalization, and successful application of processes, technologies, and tools. The documents needed to describe and substantiate the offerors' software processes and management capabilities should be listed and briefly described. As a minimum, the software processes and management related content of the SDP, SEMP, and SEMS should be described.

Briefing charts suitable for showing the overview and conduct of the SDCE are listed in table 4-11. The referenced charts are located in Volume 2, attachment 4.

Table 4-11. Bidders' Conference Briefing Outline

Vol 2, page	58	SDCE Focus
	59	SDCE Approach
	61	SDCE Role in Source Selection
	65	SDCE Model Structure
	66	SDCE Functional Areas
	67	Critical Capability Areas
	68	Critical Capability Areas (continued)
	69	Critical Capability Areas (continued)
	80	SDCE Activity Flow
	94	SDCE Proposal Data
	95	Example Cover Sheet for Project Sample Data
	96	Example Capability Definition Matrix
	97	Example Capability Implementation Matrix Showing Project Sample Data
	107	SDCE Site Visit

4.E.5.3 Prepare SDCE-Specific Input to General Notice to Offerors (GNTO)

The GNTO must explain that an SDCE site visit may be conducted prior to contract award and must provide specific information about the planning and conduct of site visits. The topics described below should be covered.

Process to be Followed in Conducting the Site Visit. The GNTO should state that a government team, members of the SSEB, may visit contractor and subcontractor facilities at which software work will be performed. It should state the expected duration of the site visit and the activities that will be conducted during the site visit, e.g., the posing of questions and examination of documentation to clarify and verify information gathered from the offeror's proposal, responses to the questions, and supporting documents. For teaming arrangements in which contractors other than the prime contractor will be performing minor portions of the software development work, it may be appropriate to conduct the SDCE site visit at the prime contractor's facility with participation by the appropriate subcontractors. However, participation by the subcontractors is at the discretion of the prime contractor. Plant and facilities tours are not required as part of the SDCE site visit.

Facilities to be Provided by the Contractor. The GNTO should list the facilities that the SDCE team will need in conducting the site visit. Main meeting rooms and breakout rooms should be specified, the number of people these rooms should be able to accommodate, the visual aid equipment needed, telephone and fax access, and the hours during which they will be required. If the team has any requirements for special access to the buildings or rooms, such as for physically handicapped members of the team or for late hour access in a secure facility, these requirements should be stated.

Types of Personnel to be Made Available. The GNTO should specify the discussion topics that are expected to be pursued during the site visit, if one is conducted. The bidder will select specific staff members to respond to the discussion topics. For instance, it would be appropriate to say that the SDCE team will wish to review configuration control. It would not be appropriate to request the presence of Mr. Reginald Smith, the CM manager.

Evaluation Approach. It should be pointed out that the site visit discussions will focus on the program at hand and the program samples (evidence of use) provided to substantiate the proposed capabilities and processes and their successful use.

Documentation to be Made Available. Additional supporting documentation will be requested by the team for examination on-site. The team will generally request the documentation prior to the visit so that it is available upon the team's arrival. The site visit agenda will include time for the examination of this documentation, but the schedule should be flexible so that, if the team should require additional time to read documentation or should discover the necessity for additional information during the site visit, these needs can be accommodated. The offeror should also have available a copy of the materials that were sent in to the government in response to the RFP.

Agenda for the Site Visit. An SDCE site visit agenda template is provided in Volume 2, attachment 2-4. It should be tailored for each specific acquisition's site visits. Site visits should be scheduled for no less than two days and for no more than four.

Schedule for the Site Visit. An individual site visit should be conducted at a time that is convenient to both the offeror and the government. However, it must be scheduled within a fairly narrow window during source selection. This window will generally open when the team has reviewed the proposals and the decision to open discussions has been made, approximately three or four weeks after the delivery of the proposals to the government (depending upon the number of offerors), and will close approximately three weeks before the SSA makes the source selection decision.

Document Security. The GNTO should assure the bidders that all data submitted by them will be stamped “Source Selection Sensitive”; that it will be maintained under strict access control in safes; that only members of the SSEB, SSAC, SDCE team, and the SSA will have access to it; and that the data will be retained by the contracting officer, destroyed, or returned to the offerors after contract award.

Debriefing. The GNTO should state that it is the intent of the government to debrief all the offerors after contract award and that, as a part of this debriefing, they will be informed of their own strengths and weaknesses in the software process and management areas as determined by the team during the conduct of the SDCE. The debriefing will be conducted at a time and date that is mutually convenient for the bidders and the government.

Volume 2, attachment 2-5 contains wording that may be used or appropriately adapted and augmented in the GNTO.

4.E.5.4 Prepare SDCE-Specific Input to Instructions to Offerors (ITO)

The ITO should specify the software process and management-related documentation that the government expects the offerors to provide with their proposals. As a minimum this documentation will include responses to the questions with substantiating documentation, and completed Cover Sheets for Project Sample Data that may be used by the government to verify the responses to the questions. The documentation should be prepared by the prime contractor or by associate or subcontractors when the latter are responsible for major portions of the software or when their software impacts the safety or security functions of the system.

Although not specifically requested for the SDCE, the proposal documentation will normally contain an SDP, SEMP, and SEMS. Whenever possible, offerors should include the SDCE information, such as responses to the questions, in these documents. Appropriate references should be provided.

The ITO should require the bidders (prime contractors) to submit substantiating data for all team members and subcontractors who will be developing significant portions of the software.

The ITO should clearly specify that several forms (the Capability Definition Matrix, the Capability Implementation Matrix, and the Cover Sheet for Project Sample Data), as shown in Volume 2, attachment 3 to this pamphlet and supplied with the RFP, should be filled out and submitted with the proposal.

The ITO should state that data submitted in direct support of the SDCE will not be counted toward any page limits for the technical and management volumes of the proposal. The ITO should also state how many copies of the SDCE data package are to be provided (typically only one). Volume 2, attachment 2-6 contains wording that may be used or appropriately adapted and augmented in the ITO.

While no specific input is required for Section M of the RFP ("Evaluation Factors for Award"), it is important that the SDCE team participate in its creation and review the final product for adequacy and consistency with respect to the SDCE material in the RFP. This section discusses the structure of the source selection, identifies where the SDCE fits among the Areas and Factors, and compares its relative importance to the other criteria for the source selection.

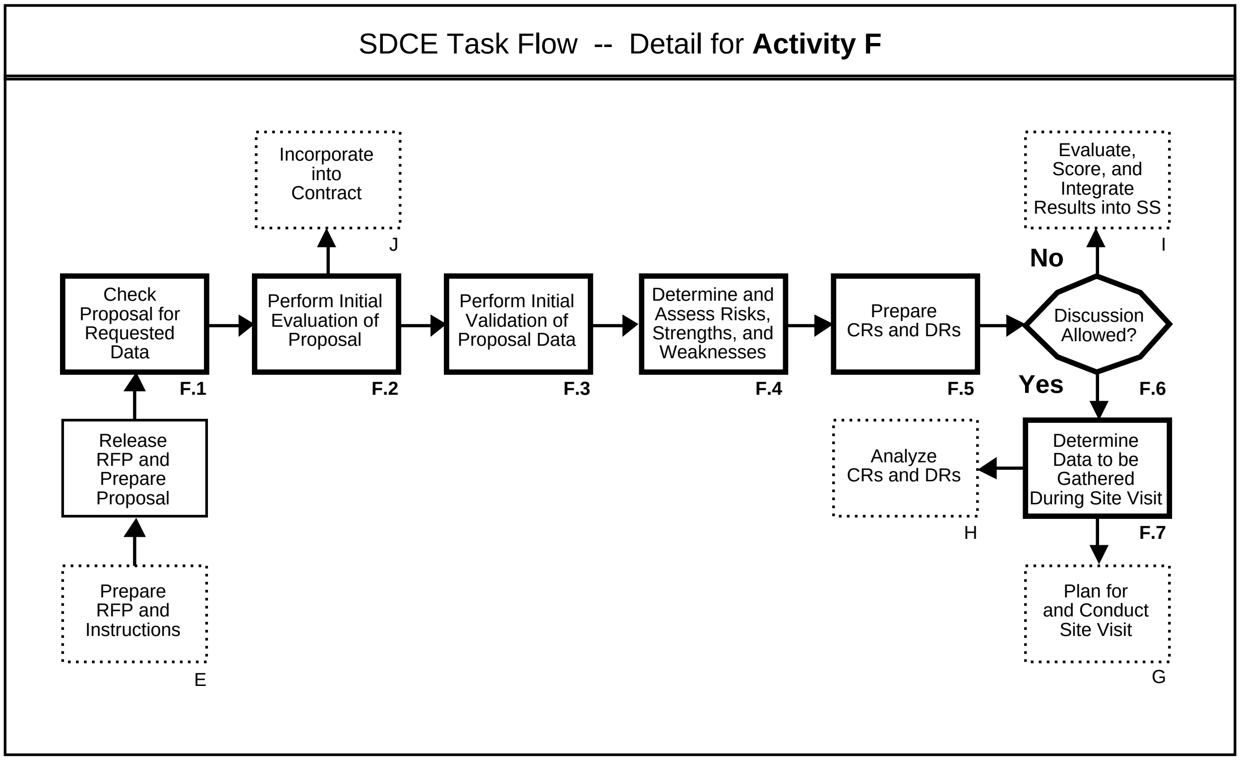
4.E.5.5 Prepare SDCE-Specific Inputs to SOW and CDRL

RFPs are usually produced by a variety of organizations and personnel. Finalization of the RFP involves, among other steps, assuring consistency between the volumes and sections of documentation that constitute the RFP and checking them for completeness and correctness. The SDCE team should be active in this finalization process to ensure that the SDCE-related material is consistent with other RFP system and software engineering requirements.

The SDCE portions of the RFP must be integrated with the other parts and checked according to source selection guidelines. For instance, the legal office and the procurement organizations should review the SDCE text portions to assure fairness and legality. The program office must assure that there will be adequate resources and time to conduct the SDCE properly. The data requested in the CDRL should be checked for redundancy and for adequacy in supplying the SDCE data as well as for assurance that the data being requested is not excessive.

The SDCE may play a key role in an incentive award fee program, depending upon its structure, intent, provisions, etc. The SDCE may be used during source selection to provide a baseline of offeror performance against which continuous process improvement may be evaluated. If the SDCE is planned to be used in an incentive award fee program, appropriate RFP coverage should be generated.

Section 4.F Review Proposals



This section provides guidance for the review of bidders' proposals to determine their software and management capability. The topics covered are:

- *How to check the proposal for the requested data*
- *How to perform an initial evaluation and validation of the proposal data*
- *How to conduct a preliminary risk assessment and determination of strengths and weaknesses*
- *How to prepare and release DRs and CRs during proposal evaluation*
- *How to determine the data to be gathered during SDCE site visits*

4.F.1 Check Proposal for Requested Data

All the volumes of a proposal must be checked to assure that the SDCE-related data requested in the RFP has been provided. The source of the data should also be considered. For example, if a major segment of the software is to be produced by a subcontractor, then data for that subcontractor and for its portion of the software should be included in the data package.

In responses to the questions, the bidders are required to supply supporting data. This data should be sufficient to satisfy the SSEB that the responses are correct, that claimed capabilities exist, that claimed technologies are indeed institutionalized, that the training courses described in the proposal are actually being taught, etc. In an acquisition in which discussion will not be allowed, it is particularly important that this documentation be complete. Examples of documentation that may be included in the SEMP, SEMS, or SDP, or that may be in separate volumes as appropriate, are described below.

SEMP/SEMS/SDP. The SEMP should clearly outline the relationship between the system engineering and the software engineering processes and how that interface will be managed. The SEMS should contain the schedule and events for the system and software development efforts and should reflect their interdependence. The SDP should define the software engineering processes that the bidder proposes to use in sufficient detail so that an evaluation may be made of the risks involved in implementing these processes. The SDP should describe, from a software perspective, the integration of software engineering activities with systems engineering activities. It should also describe the bidder's proposed software engineering organization and the resources the bidder intends to apply to the program.

Company Organization, Process, Practice, and Procedure Descriptions. The documentation on company organization and the descriptions of its processes, practices, and procedures should be checked for completeness against the requirements of the ITO. For example, if software engineering practice descriptions were requested but are missing, that should be noted. If significant information is missing from documentation or if it is unclear, a CR may be prepared to request this information or its clarification. Review criteria include:

- Company organization charts and job descriptions for staff that will be associated with the proposed program should list the job title of each individual in a position of responsibility, list him/her by name, and show his/her position in the organization and the program being bid.

- Descriptions of the software processes, practices, and procedures that are proposed to be used should be contained in the SDP and also in company manuals and standards, adherence to which is mandatory, that are made available to all the software staff. Tailoring instructions and configuration control procedures for the processes, practices, and procedures should be included in the documentation. The documentation should be clear and concise and should reflect good software engineering practices. There should be evidence that this documentation is being followed and that it is serving a useful purpose.

- Company internal standards that prescribe products and procedures that are to be used on software development projects should be detailed and unambiguous and should reflect modern software development methods. Such standards might be for corporate application of languages, for software documentation, for corporate software quality assurance procedures, for software engineering environments, etc. The software process improvement program should be described as well as its achievements and plans for its future.

Evidence of Use of Corporate Standards, Processes, Practices, and Procedures. It is a strength if the proposed standards, processes, practices, and procedures are in wide use within the offeror's software development organization. Evidence of this institutionalization should be provided.

Examples of such evidence might be minutes of meetings, lists of attendees at courses, configuration control records, SQA records, test reports, etc. For subcontractors and teaming arrangements, the value of following institutionalized practices must be weighed against the value of applying consistent practices accross the team.

Subcontractor/Teaming Management and Control Procedures. If a portion of the software will be developed by a subcontractor or team partner, the prime contractor must provide its plans to control the requirements, design, quality, and configuration of that software. The contractor should explain its management plans for outsourcing the software and for integration and testing.

Cover Sheet for Project Sample Data. For each set of project sample data which an offeror provides to support responses to the questions, the offeror must submit a completed Cover Sheet for Project Sample Data that adheres to the template in the RFP (Volume 2, attachment 3-3). This information is used by the SDCE team to determine the similarity of the programs referred to in the responses to the program being proposed.

4.F.2 Perform Initial Evaluation of Proposal Data

An initial evaluation of the software-relevant proposal data, including the offerors' responses to the questions, should be conducted after the completeness of the data has been ascertained. Steps in the evaluation include the following:

Analyze for Completeness, Ambiguities, Problem Areas, Weaknesses, Inconsistencies, and Clarity. In the steps described in subsection 4.F.1, the SDCE team checked for completeness of data on a fairly high level. In this phase, the team analyzes the data to assure that it is sufficiently complete to allow its evaluation against the evaluation standards (supported by the model criteria). The supporting data and the standard, process, practice, and procedure descriptions should be checked for completeness, ambiguities, inconsistencies, and clarity, and CRs should be prepared as required. The data should be checked for strengths and weaknesses, these should be noted, and DRs should be prepared for outstanding weaknesses and other potential problem areas.

Determine Adequacy of Information on Systems Engineering/Software Engineering Interface, Including SEMP/SEMS/SDP Relationship. The software engineering processes must be integrated with the systems engineering effort in order to provide a well engineered product. The offeror's description of this integration should be sufficiently complete that the SDCE team can make a judgment with regard to the probability of success of the proposed method. The description should encompass the entire life cycle.

Analyze the Cover Sheet for Project Sample Data for Appropriateness and Completeness. The project data contained in the Cover Sheet for Project Sample Data should be analyzed to determine the comparability of the program being acquired with those for which sample data was supplied. If programs other than comparable ones were used, the applicability of the responses must be evaluated, and a DR should be prepared if the discrepancy is extreme.

Analyze and Evaluate Question Responses and Supporting Documentation. The questions were designed to reveal each bidder's software engineering processes and management capability. The responses and the documentation must be analyzed and evaluated to determine the offeror's capabilities; these findings must then be compared with the model criteria. This is an important step in the source selection process, one in which the SDCE team will arrive at an initial assessment of each offeror's capability to produce and/or maintain the software that will be on contract. The Capability Definition Matrix and the Capability Implementation Matrix should be analyzed. CRs and DRs should be prepared as appropriate.

Compare Data to the Evaluation Standards (Supported by the Model Criteria). The analyzed data may now be used for a comparison with the evaluation criteria to determine whether each offeror meets the standards for each of the applicable factors. Details of this evaluation activity are described in section 4.I. Where data is deemed insufficient, CRs may be prepared to elicit more information. Where it is clear that an offeror does not meet the source selection evaluation standards, a DR should be written.

4.F.3 Perform Initial Validation of Proposal Data

In this task, the SDCE team uses all the data available to it to validate each offeror's claimed capability to successfully develop the proposed software on the proposed schedule and within cost, using the personnel, facilities, tools, methods, languages, hardware, environments, technologies, etc., that are described in the proposal. At this point the team should have at its disposal the bidders' responses to the questions, their supporting data, and their practices and procedures. The data for each bidder must be analyzed and then compared to the team's validation considerations. Details of this validation activity are described in section 4.I.

4.F.4 Perform Initial Assessment of Strengths, Weaknesses, and Risks

This task involves a rollup of the analyses of the previous tasks to arrive at an initial assessment of each offeror's strengths and weaknesses. This rollup could be expressed in words or in the four-color source selection scheme. If discussion is allowed, and if the team decides that this is the preferred method, this initial assessment will be followed by a site visit to each of the bidders. If, however, site visits are prohibited because of the "no discussion" rules, this is the final evaluation, and the findings should be expressed as the strengths, weaknesses, and risks of each bidder in the normal source selection color code (blue, green, yellow, and red). A separate narrative analysis and evaluation rating should be prepared for each SDCE Functional Area. A summary rating would then be prepared for the SDCE as a whole for each offeror. Details of this evaluation activity are described in section 4.I.

The team must also make a judgment about the risk to the software portion of the acquisition if the contract were to be awarded to each of the offerors. All of the risks should be considered as well as the offerors' recognition of these risks, their proposed risk avoidance and mitigation strategy, and their past history in dealing with comparable risk. One way to analyze risk is to independently evaluate risk in different ways and then combine these evaluations into a single risk assessment for each offeror.

Determine Risk of Proposed Processes, Technologies, and Risk Reduction Plans. In this evaluation, the team should ask questions such as:

- How risky is this process for this particular application, given the program's schedule and cost restraints?
- How has each offeror performed with this process under similar restraints? (Data may not always be available or comparable.)
- How adequate is each offeror's risk reduction plan? Has each recognized all the risks? Have the appropriate risk avoidance and reduction strategies been identified?
- How appropriate is the proposed technology to this application? Has it been applied before? By this offeror?

The Capability Evaluation Matrix form (Volume 2, attachment 3-4) should be used to record the evaluation for each relevant CC. The Capability Assessment portion of the CCA Score Sheet (Volume 2, attachment 3-5) should then be completed.

Determine Risk Posed by Organization and Personnel Prior Experience. This part of the risk evaluation is concerned with the people and organizations that the offerors are proposing to bring to bear on the project. Questions that should be answered by analysis of the data are:

- Is the proposed project organization appropriate for this project?
- Has the offeror used this organization successfully in a previous and similar program?
- Do the proposed personnel have adequate experience in the application, the system, the language, etc.? If not, does the offeror propose to hire or to train?
- What is the offeror's previous record in accomplishing the proposed strategy to attain the required personnel capability?

4.F.5 Prepare CRs and DRs

At each step in the proposal evaluation, CRs and DRs may be prepared as appropriate. They should be prepared regardless of the discussion/no-discussion status of the acquisition at the time. Often a no-discussion acquisition is opened to discussion midway in the source selection process when it is perceived that to do so is in the best interest of the government. CRs should be prepared for requests for clarification and for requests for missing data. They should not be prepared for trivial items but only for ones that have significance. DRs should only be prepared when a capability or process is clearly deficient with respect to the RFP requirements. The authors of DRs and CRs must be careful to preserve source selection sensitive data and must not give one contractor access to the proprietary data of another.

4.F.6 If Discussions Allowed, Release CRs and DRs

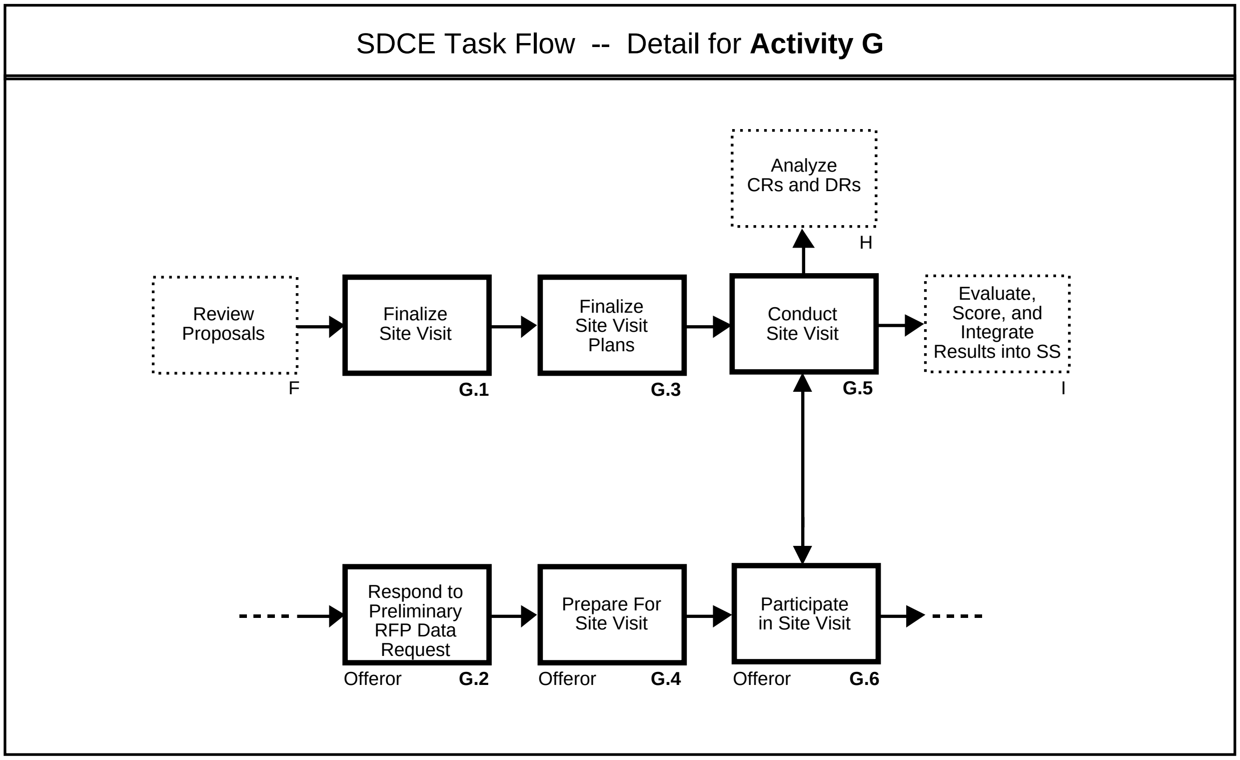
Once the decision to open discussions is made, CRs and DRs may be released. CRs and DRs must be reviewed by the SDCE Team Chairperson, Panel Chairperson, and other SSEB leaders, as required. In some instances the legal officer should also review the DRs prior to their release. When approved, the CRs and DRs should be released to the offerors with the expected response date indicated.

4.F.7 Determine Data to be Gathered via SDCE Site Visit

If, after the proposal evaluation, there are any questions about offerors' capabilities, there are two ways to obtain needed information. The first, described above, involves CRs. The second is to conduct a site visit with each bidder, during which clarifying questions may be asked and additional documentation may be examined. Two steps should be performed in preparing for site visits:

- Determine additional documentation to be requested; provide the list to each offeror.
- Prepare follow-up questions based on the analysis of the proposal information.

Section 4.G Plan for and Conduct Site Visit



The site visit is a key data gathering component of the SDCE method. It has three purposes. First, it provides a forum for the SDCE team and an offeror to discuss the proposed capability and processes, in an open dialogue, with the objective of reaching a mutual understanding of the offeror's capability in terms of processes, resources, experience, skills, tools, and technology. Second, the SDCE site visit confirms the offeror's ability to implement the software development processes and capabilities submitted in its proposal package. Finally, the SDCE site visit supports the process of constructively soliciting and agreeing to the incorporation of proposed processes in the SDP, SEMP, and SEMS. This section provides guidance to help SDCE teams appropriately and comprehensively conduct a site visit, including:

- *How to plan for the site visit*

- *How to establish appropriate communication between the PCO and the contractor point of contact regarding the site visit*

- *How to conduct a site visit*

- *How the government, contractors, and subcontractors will interact during the site visit*

- *How to handle the data utilized and generated during the site visit*

4.G.1 Evaluation Team: Plan Site Visit

As part of the source selection process for software-intensive system acquisitions, the SDCE team may conduct site visits at offerors' facilities to confirm their software development capability. To be equitable, all core members of the SDCE team should visit each of the offerors. A site visit gives the SDCE team a four-fold opportunity:

- To solicit and review additional software development capability information in support of the offeror's response to a RFP
- To access information that may be difficult to obtain in written form
- To address specific software development risk issues
- To explore capability issues in a positive, team-building atmosphere

As soon as possible, the SDCE team should identify the number and length of the site visits. A site visit typically lasts from two to four days, depending on the size of the acquisition and the evaluation of SDCE proposal data. A site visit should be conducted for each of the prime contractors bidding on a software-intensive system acquisition and may also include their subcontractors. Where contractors or subcontractors have teamed, each team member may or may not be extensively involved in software development; site visits should then be scheduled and tailored appropriately to best utilize available resources. A similar determination should be made with regard to major subcontractors to decide whether to conduct site visits at their facilities or in conjunction with the associated prime contractor. NOTE: If a site visit is conducted at a subcontractor, the prime contractor must be involved and the PCO must communicate with the subcontractor through the prime contractor. If the site visit is conducted at the prime contractor's facility with subcontractors also present (for expediency), the prime contractor may determine whether the subcontractors should participate in the prime contractor's discussions, but the prime contractor must be invited to participate during the subcontractor discussion portion.

Once the number, duration and location of site visits have been determined, the schedule and preliminary agenda can be drafted. The PCO should then notify each offeror's Defense Plant Representative Office (DPRO) to provide notification of the site visit. Immediately thereafter, a notification package should be sent to each offeror. The package should contain:

- Site visit dates, and preliminary agenda for each day
- Topics of particular interest to the SDCE team
- Clarification requests or deficiency reports the team intends to discuss
- Data the team wishes to review to validate processes described in the proposal

Follow-up questions and discussion topics should be identified by the SDCE team on the basis of the analysis of the offeror's responses to the SDCE requirements, as well as other volumes in the proposal.

A complete and clear notification package will ensure that the site visits are consist across offerors and that the best possible information will be obtained during visits. Also, notification of offerors will improve their preparation, thereby ensuring the most efficient use of the time. Figure 4-6 is an example of a site visit schedule.

4.G.2 Offeror Team: Respond to Preliminary RFP Data Request

Upon receiving notice of an SDCE site visit, the contractor identifies a point of contact to the PCO to ensure the most direct communication. This person then reachs agreement with the PCO on the dates and location of the site visit. As a minimum this should include the offeror's formal acknowledgement of the pending SDCE. It then becomes the responsibility of the prime contractor to immediately notify any involved subcontractors of their required participation.

4.G.3 Evaluation Team: Finalize Site Visit Plans

Prior to the the site visit, the SDCE team should thoroughly review all proposal data and CR and DR responses submitted by the bidder. The SDCE team will then be able to finalize the site visit schedule, agenda, and discussion topics and send them to each offeror with a follow-on cover letter. This should be done at least two weeks before the SDCE site visit. See Volume 2, attachment 2-7 for a cover letter template.

4.G.4 Offeror Team: Prepare for Site Visit

Once the agenda is finalized, the offeror must assemble the appropriate individuals to participate in the discussions during the site visit. The offeror's response team should include members of the offeror's proposal team, in particular the systems and software engineering leaders. This may include both functional representatives as well as appropriate project personnel and may include representatives of the projects identified in the example project data. By the time the site visit is conducted, the offeror will have submitted the SDCE response data with the proposal. The individuals who prepared the SDCE proposal data are the appropriate people to prepare the data required to support site visit discussion topics, as well as any discussion on CRs and DRs identified by the SDCE Team.

In preparing presentation material for the site visit, the offeror should be aware that additional consideration will not be given for elaborate briefing material. The focus should be on content. The government prefers black and white transparencies, presented on one projector. Although bound volumes are not expected, provision should be made to facilitate the SDCE team taking copies of contractor-presented material back to their home site following the site visits. Responses to SDCE questions, CRs, DRs, and discussion topics should include photocopies of referenced documents (e.g. existing proposal material, SDP, SEMS, etc.) for expediency.

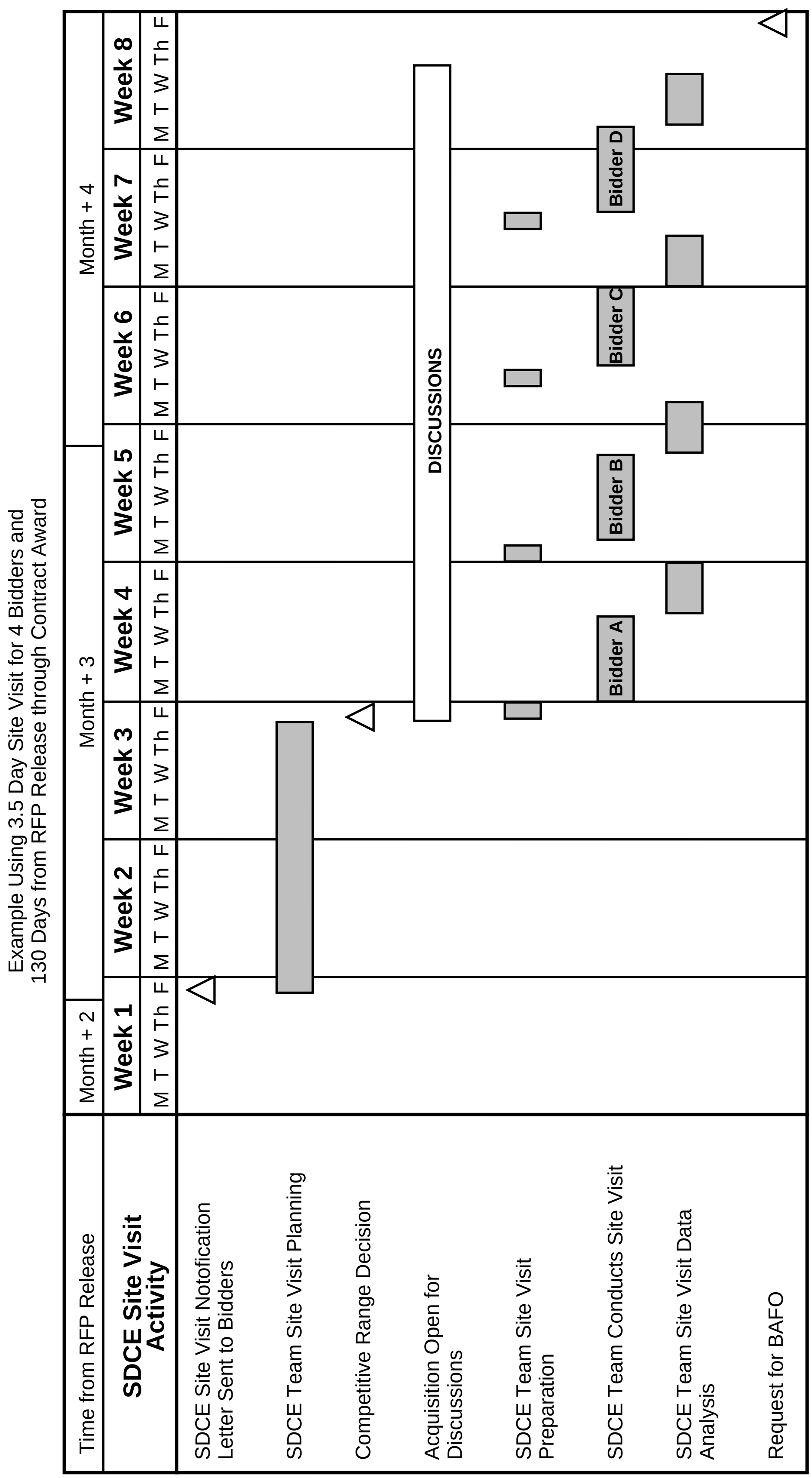


Figure 4-6. SDCE Site Visit Schedule Template

The offeror needs to make support and facility arrangements to accommodate the SDCE site visit. These should include an adequately sized conference room with telephones, fax, copying, restrooms, and refreshments nearby. Facility tours, that consume time that would be better spent in discussion, should be avoided. Similarly, secretarial support should be provided by the offeror during the visit to help with telephone messages, typing, chart presentations, and copying. The objective of thorough preparation is to minimize distractions so that all of the on-site time is focused on the SDCE discussions necessary for successful source selection.

4.G.5 Evaluation Team: Conduct Site Visit

On the first day of the SDCE site visit, the SDCE team should arrive at the designated facility in enough time to register with security and receive visitor badges prior to the start time on the agenda. Once the SDCE team and contractor's personnel arrive at the designated conference room, the SDCE team leader convenes the evaluation, introducing the evaluation team participants and identifying the organizations they represent. At this time, the SDCE Team Leader will give a presentation on the purpose of the SDCE as it relates to the subject program source selection process, will describe the SDCE method, and will answer questions the bidder may have. Suitable briefing charts are listed in table 4-12. The referenced charts are located in Volume 2, attachment 4.

Table 4-12. SDCE Site Visit Briefing Outline

Vol 2, page	110	SDCE Site Visit Overview
	111	Team Members
	80	SDCE Activity Flow
	83	SDCE Activities - Conduct
	84	SDCE Activities - Conduct (Continued)
	85	SDCE Activities - Wrap-up
	112	Integration within Source Selection Structure
	113	SDCE Site Visit

Once the SDCE method has been discussed and put in the context of the overall source selection process, the logistics of the visit should be discussed, including movement within the contractor's facility and access to the contractor software engineering community, which is at the discretion of the contractor. The agenda will be reviewed and updated, so that all participants know approximately when they will be required to provide data or to answer questions. Participant readiness is a vital element of a successfully conducted site visit. Also, the use of the CR and DR should be clearly explained. Since CRs and DRs are the official communication device used throughout source selection to obtain information and data, they require the highest possible priority by both the SDCE team and the contractor, and should be carefully logged and tracked. Contractor-prepared responses to previous issued CRs and DRs must not be given to members of the SDCE team at the site visit, but should be formally submitted to the designated source selection official to meet response time requirements. CR and DR responses become an important part of the permanent record of the source selection.

Once the logistics have been understood, the SDCE team will solicit the bidder's responses to the follow-up questions and discussion topics. SDCE site visit discussions will be conducted in a group forum, with the number of government participants being defined by the government and the number of offeror participants being defined by the offeror. Discussions with each offeror must be confined exclusively to the offeror's proposal, performance history, and identified deficiencies and clarifications. Discussions must be conducted in a way that scrupulously avoids disclosure of the relative strengths and weaknesses of competing offerors, or of the technical information, ideas, or cost (price) data from any other offeror's proposal. The SDCE team should look for completeness and adequacy in the offeror's site visit responses, as well as strong or weak level of compliance with the SDCE model criteria. SDCE team members need to ensure that they share a common understanding of the responses.

Contractor responses judged to be incomplete or inadequate should be immediately documented in the site visit notes taken by the SDCE team. The main points in these notes should be reviewed with the offeror. These notes may become the basis for CRs and DRs written during the analysis period following the site visit after proper review and approval by the SSEB leaders.

The SDCE team should keep careful records of the discussions during the site visit to be able to reference them at a later time. Volume 2, attachment 3-8 is provided to facilitate orderly recording of comments and observations. Once all responses have been received and information provided, the SDCE team then reviews any additional documentation that has been supplied in order to settle outstanding issues.

The SDCE team should take time to have final discussion among its members to assure that they fully understand the offeror's processes and capabilities and to prepare the feedback presentation, the concluding event in the SDCE site visit. The feedback presentation will be marked Source Selection Sensitive. It is imperative that all evaluation team analysis data (including documents, notes, or any other information whether in electronic or hardcopy form) be marked Source Selection Sensitive or destroyed after its use.

The SDCE team should communicate the scope of the feedback session to the offeror to assure that the offeror's expectations are consistent with what can legally be presented at this point in the source selection process. The SDCE team should be aware of the restrictions on the type of commentary and interaction that can occur during the feedback session. The guidelines shown in table 4-13, in conjunction with the use of the SDCE site visit feedback briefing template, Volume 2, attachment 2-8, will help the SDCE team prepare for and conduct the feedback session. Feedback given by the SDCE team should be documented. Detailed site visit feedback charts, with discussion restricted to the prepared data on the charts, should help to record the communicated information. Any discussion beyond the chart presentation should also be documented.

The offeror has the opportunity to respond to the SDCE team presentation, to ensure an understanding of the findings communicated by the presentation and to clarify points raised during the presentation that may have been ambigious. Prior to conclusion of the SDCE site visit, all offeror proposal data should be marked with the same legend that appears on the cover page of the offeror's

Table 4-13. Guidelines for Feedback at SDCE Site Visit

Appropriate	Not Appropriate
Objective understanding of responses, in the form of: “This is what we asked for . . .” “This is what we received from you . . .” “This is what we heard you tell us . . .”	Comparison/evaluation of results to source selection evaluation standards
	Value judgment on SDCE site visit data collected
Identification of inconsistent responses	Discussion on other offeror’s responses or success during their SDCE site visits
Identification of incomplete responses	Discussion on “a score”
Statement of success of SDCE site visit in meeting the objectives of the visit.	

proposal and handled accordingly. Data that will not be taken back to the SDCE team’s facility for further analysis should be returned to the contractor or destroyed.

4.G.6 Offeror Team: Participate in Site Visit

A contractor representative meets the SDCE team at the designated facility and is assigned for the duration of the visit to escort the SDCE team. Once the participants are assembled in the designated conference room, the offeror’s team leader will give the SDCE team a formal presentation to describe the offeror’s organization, to give an overview of the software and related systems engineering proposal, and to introduce the offeror’s team, describing the roles of each team member in the SDCE. The offeror’s presentation may also describe the facilities that are available for SDCE team use during the site visit.

The offeror should have copies of prepared responses for the follow-up questions that were sent in advance by the SDCE team. Likewise, any data that has been requested ahead of time to support CR and DR understanding should be provided and documented at this time. It is extremely helpful for the offeror to have readily available a copy of the materials that were sent to the government in response to the RFP. The offeror’s team (including subcontractors, if appropriate) should be available to answer questions and provide evidence as requested. The goal is to provide complete and accurate information to the SDCE team as quickly as possible.

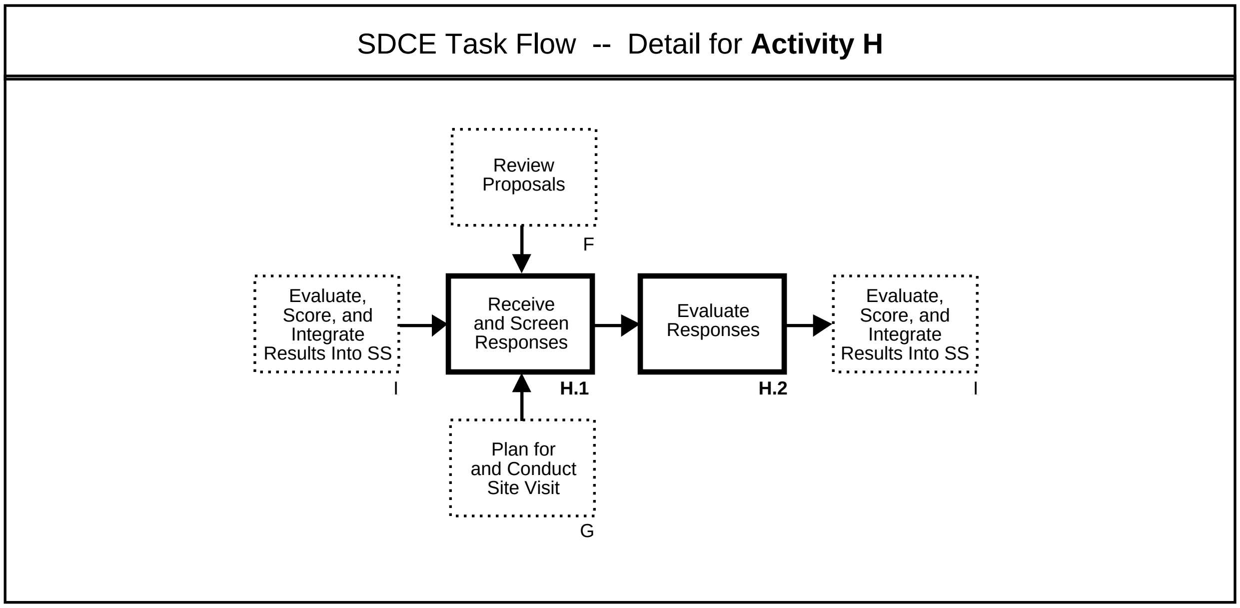
As data is provided to the SDCE team, the SDCE team and contractor must ensure that all data is marked with the legend that appears on the cover page of the offeror’s proposal. These data include documents, notes, or any other information, whether in hardcopy or electronic form.

At the conclusion of the question answering and data gathering portion of the SDCE site visit, the SDCE team will give a feedback presentation to the offeror. This presentation should convey to the offeror the SDCE team’s understanding of the offeror’s software development processes and capability. The presentation will state observations resulting from the data gathering, for final verification, understanding, and bidder comment. It is essential for the offeror to understand that the feedback presentation mirrors only the observations made during the site visit, and does not include any SDCE performance feedback. The bidder may point out factual discrepancies, and provide clarification and evidence as required.

Once the SDCE team has completed its feedback presentation, the offeror has the opportunity to ensure that the evaluation team has understood its question responses correctly. It is also incumbent upon the offeror's team to raise any contractor issues that may need to be brought to the attention of the SDCE team.

The SDCE site visit is concluded after both the SDCE team and the offeror have completed feedback of their comments in the feedback session, and made any closing remarks.

Section 4.H Analyze CRs and DRs



Clarification Requests are used in the source selection process to gain clarification of proposal material that is ambiguous or otherwise unclear in important areas. Deficiency Reports are prepared for important areas of the proposal that are determined to be deficient relative to RFP requirements. CRs and DRs generated during source selection are reviewed by the SSEB and, if discussions are opened, are submitted to the offerors for response. CRs and DRs are prepared relative to all of the offerors' proposal volumes, including the SDCE proposal material, if required and appropriate. This section focuses on CRs and DRs that are formal communication instruments used in the SDCE process, and provides the following guidance:

- *Where in the process flow CRs and DRs are generated*
- *How CRs and DRs are processed*
- *How CR and DR responses are evaluated*
- *When follow-on CRs and DRs may be needed*

4.H.1 Receive and Screen Responses

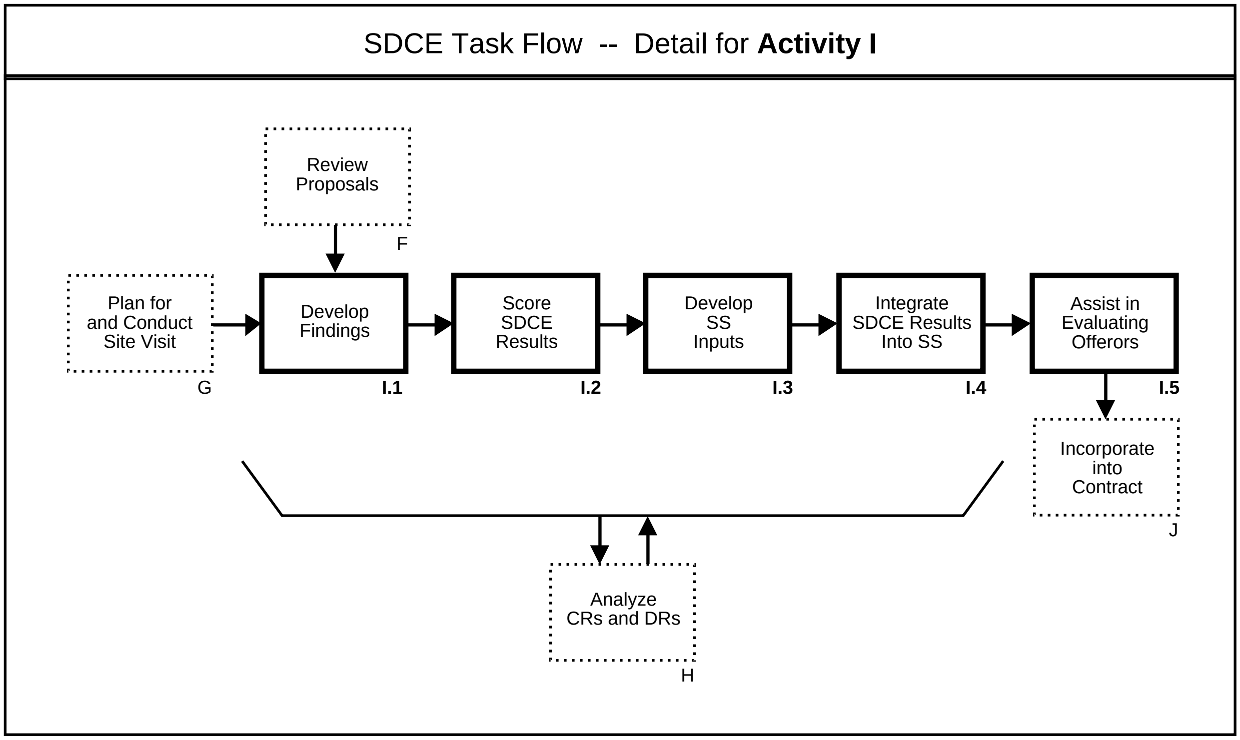
CRs and DRs may be generated during the initial proposal evaluation phase (see section 4.F), during the SDCE analysis period, and during the final phase of the source selection when the written evaluations are developed and integrated into strengths, weaknesses, and risks by Factors and Areas (see section 4.I). CRs and DRs may also be generated after the site visit based on observations noted during the visit (see section 4.G). CRs are typically written to document inconsistencies between the proposal and the site visit discussion or when written data that describes a process or capability that should go into the contractual documentation (e.g., SDP, SEMP, SEMS) is cited by the bidder during the site visit. DRs are typically written to document significant deficiencies in the offeror's site visit response. The offerors are required to respond to the CRs and DRs within a specified period of time to clarify the specific CR issues or to resolve the specific DR write-up. The SDCE team will formally review and disposition responses to SDCE CRs and DRs as part of the established source selection process.

Each response must be reviewed to determine whether sufficient information has been provided to evaluate the response against the evaluation standards. In some cases the response may provide insufficient information to determine whether the process or capability is adequate, and it may be appropriate to write an additional CR to request explicit information that was omitted. Judgment must be exercised as to whether the offeror misunderstood the original CR or the response is simply inadequate. (In this regard, care should be exercised in preparing the original CR or DR to state explicitly the information required to clarify an issue or resolve a deficiency. For example, if a CR is prepared against an offeror's sample of evidence that a process is followed, the CR should be explicit regarding the information necessary to clarify the application of the process.)

4.H.2 Evaluate Responses

Responses to CRs and DRs should be reviewed against the source selection evaluation standards and supporting model criteria. In the case of a detailed Critical Capability clarification or deficiency issue, it is necessary to evaluate the response against the criteria associated with the specific Critical Capability. In this case, the formal evaluation is performed against the source selection evaluation standard associated with the model criteria. In the case of a more general clarification or deficiency response, the evaluation against a specific model CC may not be appropriate; it may be necessary to review the response against the source selection standards for evaluation associated with designated Factors or Subfactors. Also it may be appropriate to evaluate responses to CRs and DRs in the broader and more integrated context of the scope of the proposal information and how it relates to the source selection board. For example, a response that clarifies the offeror's approach to internal design reviews in preparation for a preliminary design review may need to be coordinated with the offeror's technical proposal, preliminary SDP, and the SEMS engineering process description. In any case, responses to CRs and DRs may clarify an issue or resolve a deficiency such that a particular capability may be evaluated as a weakness, a satisfactory or adequate capability, a strength, or a capability or process risk.

4.I Evaluate, Score, and Integrate Results into Source Selection



The process of evaluating, scoring, and integrating the SDCE results into the source selection is the nucleus of the SDCE methodology. This section provides detailed guidance on the techniques and steps for performing the following critical parts of the SDCE methodology:

- *How to organize and synthesize the various SDCE data*
- *How to develop findings*
- *How to summarize or roll up the findings to higher levels*
- *How to score the findings*
- *How to develop source selection inputs*
- *How to integrate the SDCE results with the rest of the source selection*

4.I.1 Develop Findings

Development of the SDCE findings is an ongoing process starting with the initial evaluation of the proposal, continuing with the site visit review (if conducted) and post-site visit analysis and evaluation, and concluding with the evaluation of the best and final offer. The initial evaluation of the proposal produces a set of preliminary findings, consisting of the strengths, weaknesses, and risks of the offeror's approach. The preliminary findings are then used to focus the review that is conducted during the site visit. As a result of the information obtained during the site visit, the preliminary findings are refined. Development of the final findings is a further refinement that occurs as the SDCE team continues to evaluate the proposal data after the site visit, including analyses of any CR and DR responses and the offeror's BAFO.

Evaluation standards and assessment criteria are the basis for developing findings. Evaluation standards are used in conjunction with assessment criteria to measure each offeror's ability to meet the government's needs as stated in the solicitation. Evaluation standards establish the minimum level the offeror must meet to be judged acceptable. Assessment criteria are guidelines that help the source selection evaluators identify strengths, weaknesses, and risks. Assessment criteria typically address aspects such as soundness of approach, understanding of the requirement, and compliance with the requirement.

4.I.1.1 Synthesize Data

The evaluation of an offeror's software development capability is based on many different sources of information. The initial data submitted with the proposal includes a wide range of source material, including the SEMP, SEMS, and SDP; company processes, procedures, and standards; project sample data showing evidence of use; and the answers to the SDCE questionnaire. Additional information may be gathered during site visit discussions, in responses to CRs and DRs, and in the BAFO. The evaluation cannot be based on any one of these sources of information in isolation. All the information must be synthesized in order to develop an integrated picture of the offeror's overall software development capability. At any given time during the development of findings, different aspects of the evaluation may be in various states of progress, and the analysis of different offerors' proposals may be in various stages of completion. It is important, therefore, that when considering a particular topic as a whole, an integrated view of that topic be achieved.

4.I.1.2 Determine Strengths and Weaknesses

To determine strengths and weaknesses, the evaluation team must judge the adequacy of the proposed approach for each Critical Capability. The adequacy of the approach is determined by assessing how well the offeror meets evaluation and validation considerations. The evaluation considerations address the soundness, goodness, or quality of the proposed approach. The validation considerations address the ability of the offeror to successfully implement the approach. The evaluation and validation considerations for each CC are rated as either strong, acceptable, or weak. The guidelines shown in table 4-14 are from Air Force FAR Supplement Appendix AA (AFAC 92-33) and should be used in determining strengths and weaknesses. "Acceptable" is defined as anything that is neither strong nor weak. The strong, acceptable, and weak ratings are documented on a Capability Evaluation Matrix (see figure 4-11).

4.I.1.2.1 Assess Evaluation Considerations for Strengths and Weaknesses (figure 4-11)

The first step of developing findings is to examine the adequacy of the offeror's proposed approach by assessing the SDCE evaluation considerations. The evaluation considerations, which address the soundness or goodness of the offeror's software development approach, are discussed below.

Table 4-14. Strength and Weakness Guidelines

Rating	Description
Strength	"A significant, outstanding, or exceptional aspect of an offeror's proposal that, in the evaluation team's judgment, exceeds the minimum program requirment and evaluation standard and provides a useful capability that will be included in the specification or statement of work, or is inherent in the offeror's process, so that the government will be assured of receiving the benefits under the resultant contract."
Weakness	"An aspect of or omission from an offeror's proposal that contributes to a deficiency in meeting an evaluation standard or is otherwise a shortcoming of the proposal that has the potential to degrade contract performance."

Proposed Approach Meets the SDCE Model Criteria. How well the offeror's approach satisfies the SDCE model is the most important consideration the evaluation team must assess. Each CC has an associated set of SDCE model criteria against which the offeror's proposed approach is evaluated. The answers to the SDCE questionnaire along with the supporting documented processes, procedures, and standards are compared to the SDCE model criteria. Based on this comparison, the SDCE team must determine the degree of conformance of the offeror's proposal with each criterion. The level of compliance to the criteria and the relative importance of the criteria to the proposed project are then used to rate the proposed approach as strong, acceptable, or weak. General guidelines for developing a SDCE model rating are shown in table 4-15.

Table 4-15. Capability Assessment Guidelines

Rating	Description
Strong	Conformance to a majority of the model criteria is strong and there are no significant areas of nonconformance.
Acceptable	Conformance to a majority of the model criteria is acceptable and there are no significant areas of nonconformance.
Weak	There are significant areas of nonconformance to the model criteria.

Compatibility Among Team Members is Demonstrated. If prime contractor/subcontractor or teaming relationships are present, the evaluation team must assess if the proposed capability is compatible with the capabilities of the other partners. The approaches need not be identical but must provide an overall consistent capability. The SDCE team can determine compatibility by comparing the capability descriptions of the prime contractor, team members, and subcontractors that are submitted with the proposal. Based on the level of compatibility, the offeror's approach is rated as either strong, acceptable, or weak.

Proposed Approach is Consistent With Other Volumes of the Proposal. The information provided by the offeror to support the SDCE should agree with the information contained in other sections of the proposal. The evaluation team must therefore review the other sections of the proposal to ensure information and approaches are consistent. Inconsistencies should be resolved through

CRs. Based on the level of consistency within the proposal, the offeror's approach is rated as strong, acceptable, or weak.

Responsibility for the Proposed Approach is Identified. The evaluation team must assess how specifically and completely the offeror has identified responsibility for implementing the proposed capability. Responsibility may be identified by organizational element (test group), position title (chief engineer), or in some other acceptable fashion. Based on how well responsibility for the capability is identified, the approach is rated as strong, acceptable, or weak.

Proposed Approach is Documented in Contractual Vehicles (SDP, SEMP, SEMS). The offeror's level of commitment to implement the proposed approach is determined by assessing how well the approach is described in contractual vehicles like the SDP, SEMP, and SEMS. The Capability Definition Matrix (see figure 4-7) helps the evaluation team assess the level of commitment by graphically depicting how well the proposed capabilities are integrated into the SDP, SEMP, and SEMS. The Capability Definition Matrix is completed by the offeror and is submitted with the proposal. Under the "Commitment" section of the matrix, the offeror indicates for each CC the contractual vehicles that contain that capability. The "Comments" section on the matrix also lists the location within the SDP, SEMP, and SEMS where processes are documented.

It's not appropriate to document all proposed capabilities in the SDP, SEMP, or SEMS. The SDCE team's primary focus should be to ensure that process-related capabilities are incorporated into the

CAPABILITY DEFINITION MATRIX

OFFEROR A

FA Quality Management and Product Control

CCA Software Configuration Management

	COMMITMENT			INSTITUTIONALIZATION				LOCATION OF CAPABILITY DESCRIPTION
CRITICAL CAPABILITIES:	SOFTWARE DEVELOPMENT PLAN	SYSTEMS ENGINEERING MASTER PLAN	SYSTEMS ENGINEERING MASTER SCHEDULE	COMPANY PROCEDURE OR STANDARD	PROJECT PROCEDURE OR STANDARD	OTHER SUPPORTING MATERIAL	PROPOSAL AND SDCE QUESTIONNAIRE	LIST THE TITLES OF REFERENCE DOCUMENTS AND THE LOCATION OF THE DESCRIPTION WITHIN THE DOCUMENT
SCM Planning		X	X		X			SEMP Pg 24; SEMS Pg 8; ABC Corp CM Procedures Pg 2-11; Draft Project CM Plan Pg 3-11, 17-21, 26, 31-33
Baseline/Configuration Identification and Management	X			X	X			SDP Pg 11; ABC Corp CM Procedures Pg 12-14, 23-27; Draft Project CM Plan Pg 13-16, 21-22, 28
Configuration Audits				X	X			ABC Corp CM Procedures Pg 36-37; Draft Project CM Plan Pg 36-40
Configuration Control and Status Accounting				X	X			ABC Corp Procedures Pg 11-13, 15-19; Draft Project CM Plan Pg 12-18, 22-27
Configuration Management Library and Tools	X	X	X		X	X	X	SDP Pg 28-31; SEMP Pg 26-27; SEMS Pg 14; Draft Project CM Plan Pg 38-41; Library Procedures Pg All; Proposal Pg 86

Version 1.0   Place an "X" in every column that applies   Page 25 of 33 Pages

Figure 4-7. Example Capability Definition Matrix

appropriate contractual vehicles. Based on the level of commitment in contractual vehicles to implement the proposed approach, the CC is rated as strong, acceptable, or weak.

4.I.1.2.2 Assess Validation Considerations for Strengths and Weaknesses (figure 4-11)

The second step of developing findings is to examine the likelihood the offeror will successfully implement the proposed approach by assessing the offeror's ability to satisfy the SDCE validation considerations. These validation considerations are discussed below.

Proposed Approach is Adequately Defined and Documented. To be adequately defined and documented, the proposed approach must be understandable, defined to the level of detail needed for implementation, and institutionalized. The Capability Definition Matrix and the processes, procedures, and standards provided by the offeror are used to determine how well the proposed approach is defined and documented. The Capability Definition Matrix provides a quick means of determining how and where the capability is documented. The "Institutionalization" section on the matrix depicts how the capability is documented. In this section, the offeror records whether the proposed capability is a company standard, is a project standard, is defined in some other document, or is only defined in the proposal and questionnaire responses. The categories are listed in descending order of precedence, with the "company standard" showing the highest level of institutionalization and the "proposal only description" the least amount of institutionalization. The "Comments" section on the matrix lists the document name and page number where the capability is described. In addition to determining the level of institutionalization, the evaluation team must review the capability documentation to determine if the proposed approach is understandable and is defined to the level of detail needed for implementation. Based on the adequacy of the definition and the level of institutionalization, the proposed approach is rated as strong, acceptable, or weak.

For New Capabilities, an Adequate Analysis of Benefits Versus Risks is Provided. For new capabilities, the offeror must demonstrate that the benefits outweigh the risks. A new capability is defined as any area where the offeror does not have an established track record of employing the capability on other projects. To evaluate new capabilities, the SDCE team must assess if an adequate analysis of the benefits versus the risks has been conducted, if the risks of using the new capability are known, and if the offeror has a strategy for managing those risks. The "New Capability" column on the Capability Implementation Matrix (see figure 4-8) identifies the new capabilities being proposed. To help the evaluation team assess the new capabilities, the offeror provides a rationale for introducing the new capability on the proposed program and describes its approach for introducing and managing the new capability. The offeror's risk assessment submitted with the proposal can also be used to determine how well the risks associated with the new capability are identified and the offeror's strategy for managing them. In evaluating new capabilities, the SDCE team must be aware that the government sometimes forces offerors to use new processes. In these cases, the evaluation can only be based on the offeror's approach for introducing and managing the new capability. Based on the offeror's understanding of the risks and potential benefits and the offeror's approach for introducing and managing the new capability, the proposed approach is rated as either strong, acceptable, or weak.

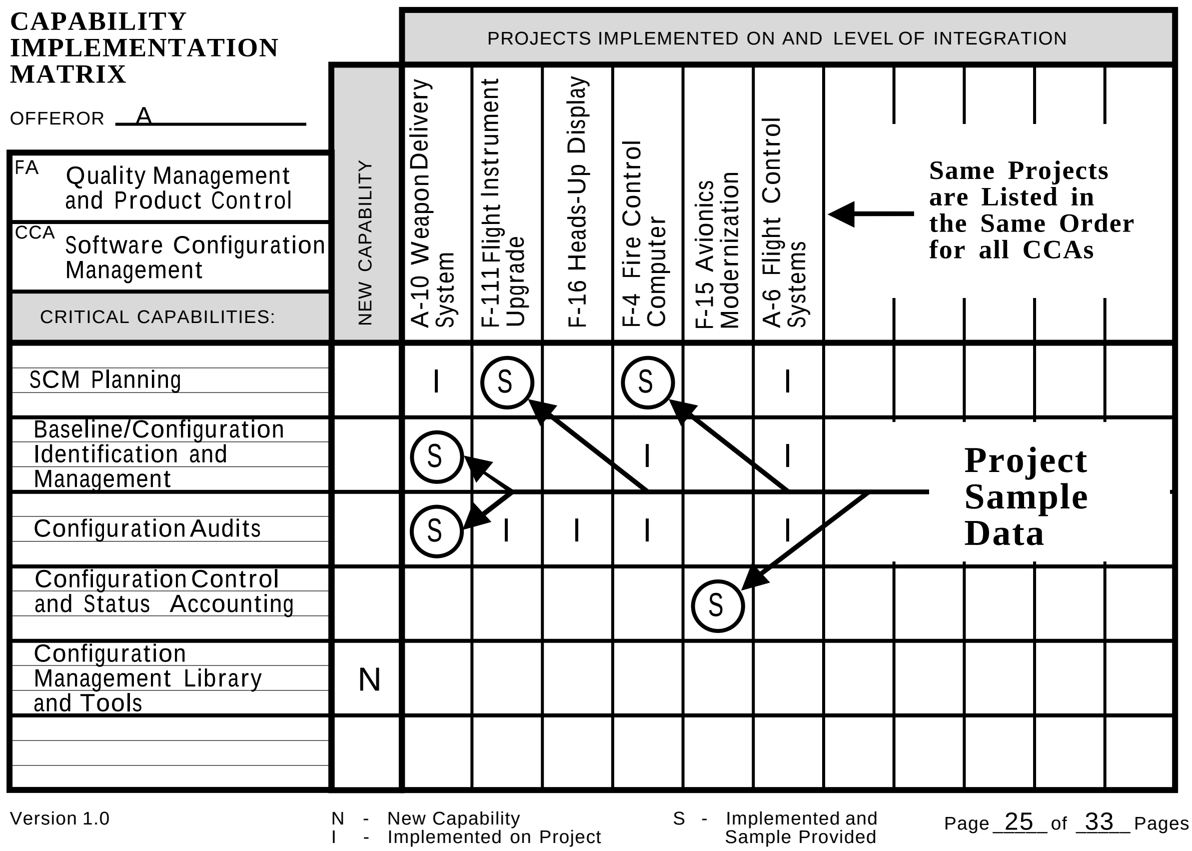


Figure 4-8. Example Capability Implementation Matrix Showing Project Sample Data

Project Samples are Relevant and Demonstrate Application of the Capability. The SDCE team must make an evaluation of the offeror’s present and past work record to assess confidence in the offeror’s ability to successfully perform as proposed. The offeror’s historical ability to implement the proposed approach for each CC is documented on the Capability Implementation Matrix (see figure 4-8) and in project sample data.

The Capability Implementation Matrix is completed by the offeror and is returned with the proposal. The “Projects Implemented on and Level of Integration” section of the matrix shows the offeror’s level of experience with the capability and lists the sample data submitted along with the projects they were used on. By reviewing the Capability Implementation Matrix and the project sample data, the evaluation team can assess the historical ability of the offeror to implement the proposed capability.

The offeror must also demonstrate that the sample data is relevant to the proposed project. Relevancy is determined in two ways. First, is the project from which the sample data was taken similar to the proposed project? Second, is the sample capability similar to the proposed capability? To help the evaluation team determine if the sample project data is relevant to the proposed project, the offeror fills out a Cover Sheet for Project Sample Data (see figure 4-9) and attaches it to the front of each project sample that is submitted. The cover sheet lists a number of project attributes (like application domain, software team size, and language used) for the proposed project and for the sample project. A blank row at the bottom of the cover sheet may also be used by the SDCE team to specify an additional critical project-specific attribute. By comparing the attributes of the proposed project to

Cover Sheet for Project Sample Data

Offeror: A

Sample Project Name: B-1B Navigation and Weapons Upgrade

Title of Sample: Cost Estimation Worksheets

Critical Capability: Software Estimating

ATTRIBUTES	PROPOSED PROJECT	SAMPLE PROJECT
Application Domain	Fighter Aircraft	Bomber Aircraft
Product Type	Avionics	Navigation and Weapon Delivery
Acquisition Phase	EMD	EMD
Software Development Phase	Requirements Analysis	Integration Testing
Award Date		March 1989
Contract Duration	9 Years	6 Years
Subcontractors	3 Software Subs	1 Software Sub
Software KSLOC	1200 KSLOCs	200 KSLOC
Software Team Size	260	65
Language(s) and Percentage	Ada (100%)	Jovial (85%) Assembly (15%)
Target Processor(s)	R3000 MIL-STD-1750	MIL-STD-1750
Applicable MIL-STDs	MIL-STD-2167A MIL-STD-2168	MIL-STD-2167

Version 1.0

Figure 4-9. Example Cover Sheet for Project Sample Data

the sample project, the evaluation team can determine how relevant the sample data is to the proposed project. The evaluation team must also compare the content of the project sample data to the description and documentation for the proposed capability to ensure they are similar and therefore relevant.

Based on how relevant the project sample data is, and on the level of historical ability to implement the capability, the proposed approach is rated as strong, acceptable, or weak.

Capability has been Integrated with Other Proposed Capabilities. Another critical consideration the evaluation team must assess is the ability of the offeror to successfully combine the individual CCAs/CCs into an integrated software development capability. The objective of this step is to evaluate the compatibility of the CCAs/CCs and the extent to which they have been integrated (used together) in the past. The historical level of integration of the capabilities is documented on the Capability Implementation Matrix (see figure 4-10) under the "Projects Implemented on and Level of Integration" section. This section is filled out by the offeror and lists the projects from which sample data was taken as well as any other projects the offeror wants to include to demonstrate past

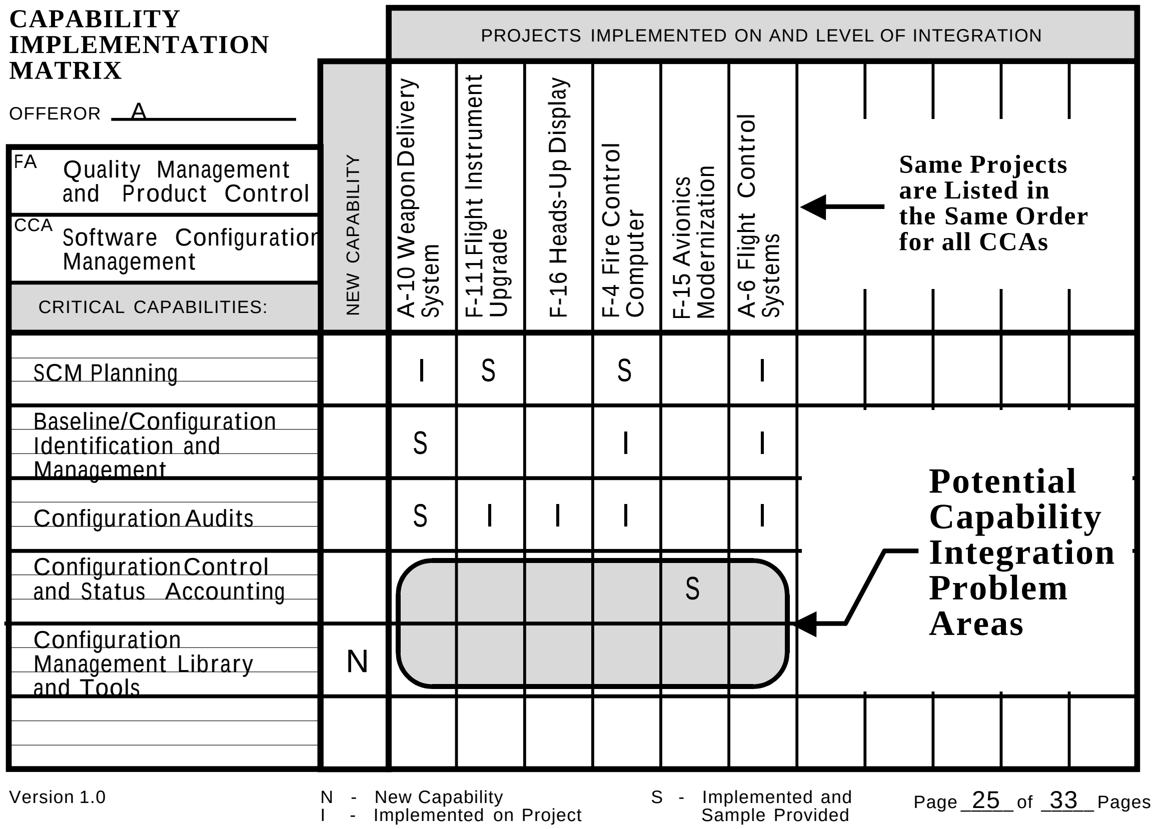


Figure 4-10. Example Capability Implementation Matrix Showing Integration Problem Areas

integration of the CCAs/CCs. The letter "I" indicates the proposed approach for each CC was implemented on that project. The letter "S" indicates the CC was implemented on the project and a sample was also provided.

By examining the implementation matrix, the evaluation team can assess how well the CCAs/CCs have been integrated in the past. The matrix provides a visual indication of where CCAs/CCs have and have not been previously used together. Areas where proposed capabilities have not been used together indicate potential capability integration problem areas. Once these areas are identified, the SDCE team must evaluate their criticality. Not all CCAs/CCs need to be closely coupled. If the CCA/CC is relatively independent of the other CCAs/CCs, the lack of previous integration is not significant. If, however, a review of the capability definitions and documentation shows that the CCA/CC has critical interfaces to and dependencies with other CCAs/CCs, a lack of previous integration may represent a capability integration problem area. Based on the level of past integration of the CCAs/CCs and the magnitude of potential integration problem areas, the proposed approach is rated as strong, acceptable, or weak.

4.I.1.3 Identify Risks (figure 4-11)

Two types of risk assessments are conducted during source selection: proposal risk and performance risk. Proposal risk entails the identification and assessment of the risks associated with an offeror's proposed approach as it relates to accomplishing the requirements of the solicitation. Performance risk entails the assessment of an offeror's present and past work record to assess confidence in the offeror's ability to successfully perform as proposed. Performance risk is assessed by the Performance Risk Assessment Group (PRAG). As a result, the SDCE evaluation concentrates on assessing proposal risk. However, the line between proposal risk and performance risk is a fuzzy one. As part of assessing proposal risk, the SDCE team must evaluate the ability of the offeror to implement the proposed approach. To do this, the SDCE team examines sample data to validate that the offeror has applied the approach on past or current programs and reviews past applications to substantiate proposed processes and capabilities.

Proposal risks that must be assessed are those associated with cost, schedule, performance, and logistics/supportability aspects of the program. Risks may be inherent in a proposed approach by virtue of its relationship to the state of the art. Risks may occur as a result of the selection of a particular approach (technical, schedule, processes, tools, etc.). Risks may also result from the prime contractor's subcontract arrangements. Furthermore, risk may occur from the offeror's ability to implement the approach. To determine proposal risk, the SDCE team must assess three risk considerations for each CC:

- What is the probability of failure?
- What is the impact of the failure?
- How easily can the deficiency be corrected?

These risk considerations are a type of assessment criteria. The considerations are rated as high, moderate, or low risk. The high, moderate, and low ratings are documented on a Capability Evaluation Matrix (see figure 4-11).

Three sources of information are available to the SDCE team to help assess risk. The first is the program office risk assessment completed prior to releasing the RFP. The second source is the risk assessment completed by the offeror. As part of the proposal, the offeror is required to submit a proposal risk analysis which identifies proposal risk areas and recommended approaches to minimize the impact on program success. The third source is the SDCE team's assessment of the evaluation and validation considerations. The strengths and weaknesses of the offeror's approach and the offeror's ability to implement the approach are indicators of high and low risk areas. These three sources of information on proposal risk must be synthesized in order to assess the risk level.

Assess Probability of Failure. The probability of failure is based on the soundness of the offeror's approach and on the offeror's capability to implement that approach. The likelihood of failure is determined by examining the risk assessments developed by the program office and offeror and the level of compliance to the evaluation and validation considerations. Initial risk assessments from the program office and offeror are available for the SDCE team to review. The adequacy of the offeror's risk assessment should be evaluated against the following criteria:

- The risk assessment provides a detailed examination of the program to uncover potential risk areas.
- The risk assessment shows an effective understanding of the program's potential risk areas.

The strengths and weaknesses identified for the evaluation and validation considerations is another basis for calculating the probability of failure. An offeror whose capability does not match the proposal requirements can increase the risk inherent in the software development effort. Conversely, a highly experienced offeror with well institutionalized processes, trained personnel, appropriate tools and facilities has a better chance to successfully overcome program difficulties. For example, if the proposed approach is strong in all the evaluation and validation consideration areas, then the probability of failure would decrease. In contrast, if the proposed approach is weak in all the evaluation and validation consideration areas, then the probability of failure would increase. Based on the adequacy of the offeror's risk assessment and the level of compliance to the evaluation and validation considerations, the proposed approach is rated as high, moderate, or low risk.

Assess Potential Impact of Failure. The second aspect of risk is the seriousness of the consequences. The SDCE team must assess the potential level of impact to the program from the risk area. Program impact must be evaluated in two ways. First, what is the impact if the offeror is proposing to implement a risky approach? Second, what is the impact if the offeror fails to successfully implement a good approach? Based on the magnitude of the potential impact to the program, the approach is rated as high, moderate, or low risk. The proposal risk impact ratings shown in table 4-16 are consistent with the risk definitions in AFFARS AA.

Table 4-16. Risk Impact Ratings

Rating	Definition
High	"Likely to cause significant serious disruption of schedule, increase in cost, or degradation of performance...."
Moderate	"Can potentially cause some disruption of schedule, increase in cost, or degradation of performance."
Low	"Has little potential to cause disruption of schedule, increase in cost, or degradation of performance."

Assess Correctability of the Deficiency. The third aspect of risk is how easily the deficient area can be corrected. To assess the correctability of the risk area, the SDCE team should consider the following:

- Are the offeror's proposed risk handling approaches adequate?
- Are there improvement activities underway which would correct the deficiency?
- Are alternative approaches available?

Based on the potential to avoid or abate the deficiency, the approach is rated as either high, moderate, or low risk. Note that a high rating here refers to high risk and not a high level of correctability. The rating really refers to the inability of the offeror to correct the deficiency. This approach is taken to keep the rating consistent with the ratings for the other two risk considerations. The risk consideration listed on the capability evaluation matrix also refers to the inability of the offeror to correct the deficiency and not the correctability of the deficiency.

4.I.1.4 Develop Capability and Risk Assessments

Using its collective professional judgment and a consensus decision making process, the SDCE team must develop a capability and risk assessment for each CC.

Record Findings. The Capability Evaluation Matrix (see figure 4-11) is a worksheet used by the SDCE team to document the detailed results of the evaluation; it provides a standard structure within which all the SDCE team's appraisals can be consolidated. By providing a standard format in which to document findings, the matrix allows team members to easily compare individual evaluation results with each other. All the key considerations assessed during the SDCE evaluation are located on the matrix, thus limiting the need to shuffle through multiple documents.

CAPABILITY EVALUATION MATRIX

OFFEROR A

FA Quality Management and Product Control

CCA Software Configuration Management

CRITICAL CAPABILITIES:	EVALUATION CONSIDERATIONS					VALIDATION CONSIDERATIONS				RISK CONSIDERATIONS		
	Compatibility of Approach Among Team Members and Primes/Subs is Demonstrated	Proposed Approach is Consistent With the Other Volumes of the Proposal	Responsibility for the Proposed Approach is Identified	Proposed Approach is Documented in Contractual Vehicles (SDP, SEMP, SEMS)	Proposed Approach Meets the SDCE Model Criteria	Proposed Approach is Adequately Defined and Documented	For New Capabilities, an Adequate Analysis of Benefits vs Risks is Provided	Samples are Relevant and Demonstrate the Application of the Capability	This Capability has Been Integrated With Other Proposed Capabilities	Probability of Failure for the Proposed Approach	Potential Impact of Failure for the Proposed Approach	Correctability of the Deficiency of the Proposed Approach
SCM Planning	S	A	S	S	S	S	NA	S	A	L	H	L
Baseline/Configuration Identification and Management	W	A	A	A	W	A	NA	A	A	H	H	M
Configuration Audits	A	A	S	NA	A	A	NA	A	W	L	L	L
Configuration Control and Status Accounting	A	A	A	NA	A	W	NA	W	A	M	H	M
Configuration Management Library and Tools	W	NA	W	S	S	W	A	NA	W	L	M	M

Version 1.0 S - Strong A - Acceptable W - Weak H - High M - Moderate L - Low NA - Not Applicable NE - Not Evaluated Page_____of _____Pages

Figure 4-11. Example Capability Evaluation Matrix

The capability evaluation matrix is divided into three parts: evaluation considerations, validation considerations, and risk considerations. Each consideration is assessed at the CC level, and its rating is recorded on the matrix. The evaluation and validation considerations are normally rated as Strong (S), Acceptable (A), or Weak (W). The risk considerations are normally rated as High (H), Moderate (M), or Low (L). However, any consideration can be rated as Not Applicable (NA), or Not Evaluated (NE). A Not Applicable rating signifies the capability is not relevant to the offeror's proposed approach. A Not Evaluated rating signifies the SDCE team chose not to look at that capability area.

The analysis that is documented on the Capability Evaluation Matrix is dependent on data from a number of different sources. Figure 4-12 shows how these different sources of information feed the assessment of the individual evaluation, validation, and risk considerations. Figure 4-13 also shows how the information contained on the Capability Definition Matrix, Capability Implementation Matrix, and the Cover Sheet for Project Sample Data relate to the Capability Evaluation Matrix.

Develop an Overall Capability Assessment for each CC. To develop an integrated assessment of the offeror's capability, the SDCE team must merge the individual evaluation and validation consideration results into a single capability rating for each CC.

The evaluation and validation considerations may have a mixture of strong, acceptable, and weak ratings (see figure 4-14). In addition, the individual considerations will have varying levels of significance to the proposed program. Moreover, considerations like the SDCE model criteria and past application of the capability are generally much more significant than other considerations. For all these reasons, there is no algorithm for combining the individual consideration results into a single capability assessment rating. The SDCE team must rely upon its professional judgment and take into account the importance of each consideration to the proposed program. General guidelines for developing an overall capability assessment for each CC are shown in table 4-17.

Table 4-17. Capability Assessment Guidelines

Rating	Description
Strong	The majority of the evaluation and validation considerations are strong there are no significant weak considerations.
Acceptable	The majority of the evaluation and validation considerations are acceptable and there are no significant weak considerations.
Weak	There are significant weak evaluation and validation considerations.

Once the capability assessment rating is established for the CC, it is recorded in the "Capability Assessment" column on the CCA Score Sheet (see figure 4-15). In addition, short descriptions of the strengths and weaknesses should also be recorded in the "Comments" section of the score sheet. It is very important for the SDCE team to capture descriptions of the strengths and weaknesses, since they will be used later to support the narrative write-up of the SDCE findings.

Develop an Overall Risk Assessment for each CC. To develop an overall risk assessment for each CC, the SDCE team must combine the results of the separate risk considerations into a single risk assessment rating (see figure 4-16). Proposal risk is a function of the probability of failure, the impact of the failure, and the correctability of the deficiency. All these risk considerations must be taken into account in developing an overall risk assessment.

The risk considerations are individually rated as high, moderate, or low. As with the capability assessment, the SDCE team must use its professional judgment to combine the individual risk consideration results into a single risk assessment rating. Once the risk assessment rating is

CAPABILITY EVALUATION MATRIX

OFFEROR A

FA Quality Management and Product Control

CCA Software Configuration Management

CRITICAL CAPABILITIES:	EVALUATION CONSIDERATIONS: Compatibility of Approach Among Team Members and Primes/Subs is Demonstrated	EVALUATION CONSIDERATIONS: Proposed Approach is Consistent With the Other Volumes of the Proposal	EVALUATION CONSIDERATIONS: Responsibility for the Proposed Approach is Identified	EVALUATION CONSIDERATIONS: Proposed Approach is Documented in Contractual Vehicles (SDP, SEMP, SEMS)	EVALUATION CONSIDERATIONS: Proposed Approach Meets the SDCE Model Criteria	VALIDATION CONSIDERATIONS: Proposed Approach is Adequately Defined and Documented	VALIDATION CONSIDERATIONS: For New Capabilities, an Adequate Analysis of Benefits vs Risks is Provided	VALIDATION CONSIDERATIONS: Samples are Relevant and Demonstrate the Application of the Capability	VALIDATION CONSIDERATIONS: This Capability has Been Integrated With Other Proposed Capabilities	RISK CONSIDERATIONS: Probability of Failure for the Proposed Approach	RISK CONSIDERATIONS: Potential Impact of Failure for the Proposed Approach	RISK CONSIDERATIONS: Correctability of the Deficiency of the Proposed Approach
SCM Planning	S	A	S	S	S	S	NA	S	A	L	H	L
Baseline/Configuration Identification and Management	W	A	A	A	W	A	NA	A	A	H	H	M
Configuration Audits	A	A	S	NA	A	A	NA	A	W	L	L	L
Configuration Control and Status Accounting	A	A	A	NA	A	W	NA	W	A	M	H	M
Configuration Management Library and Tools	W	NA	W	S	S	W	A	NA	W	L	M	M
	A, B, D, I	A, C	A, B, D	F, G, L	A, B, D, K	B, D, L	D, H, J	D, E, N	A, B, M	O, P, Q		

Sources of Evaluation Information:

A. SDCE Questionnaire Responses

B. Documented Processes, Procedures, Practices, and Standards

C. Proposal

D. Site Visit

E. Sample Evidence of Use

F. SEMP/SEMS

G. Software Development Plan

H. Rationale for New Processes

I. Plans for Teaming/Subcontracting

J. Process Improvement Description

K. Model Criteria

L. Capability Definition Matrix

M. Capability Implementation Matrix

N. Cover Sheet for Project Sample Data

O. Program Office Risk Assessment

P. Offeror's Risk Analysis

Q. Identified Strengths and Weaknesses

Figure 4-12. Sources of Information for Developing Capability and Risk Assessments

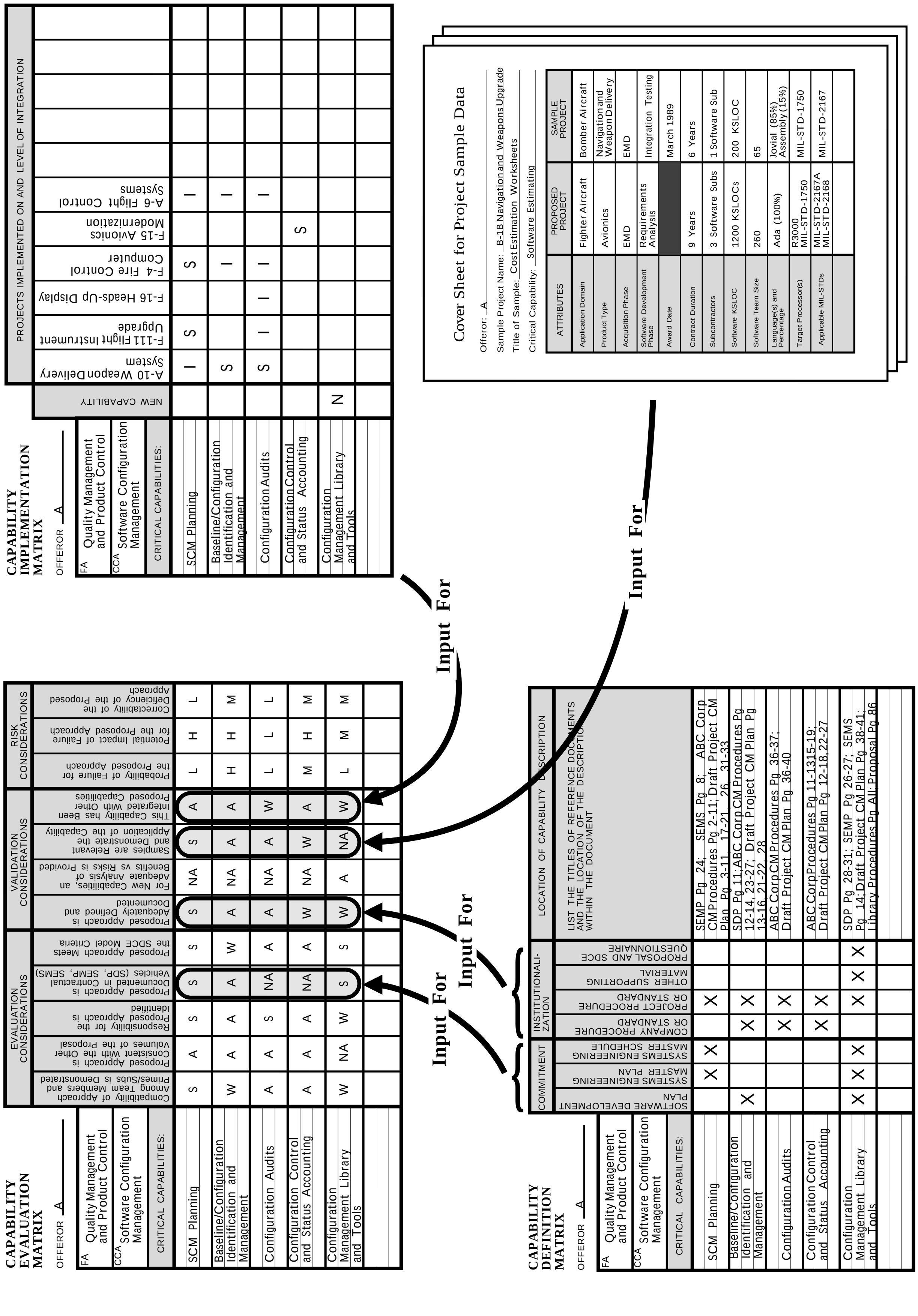


Figure 4-13. Relationship of Offeror-Completed Forms to Capability Evaluation Matrix

CAPABILITY EVALUATION MATRIX

OFFEROR A

FA: Quality Management and Product Control

CCA: Software Configuration Management

CRITICAL CAPABILITIES:	EVALUATION CONSIDERATIONS: Compatibility of Approach Among Team Members and Primes/Subs is Demonstrated	Proposed Approach is Consistent With the Other Volumes of the Proposal	Responsibility for the Proposed Approach is Identified	Proposed Approach is Documented in Contractual Vehicles (SDP, SEMP, SEMS)	Proposed Approach Meets the SDCE Model Criteria	VALIDATION CONSIDERATIONS: Proposed Approach is Adequately Defined and Documented	For New Capabilities, an Adequate Analysis of Benefits vs Risks is Provided	Samples are Relevant and Demonstrate the Application of the Capability	This Capability has Been Integrated With Other Proposed Capabilities	RISK CONSIDERATIONS: Probability of Failure for the Proposed Approach	Potential Impact of Failure for the Proposed Approach	Correctability of the Deficiency of the Proposed Approach
SCM Planning	S	A	S	S	S	S	NA	S	A			
Baseline/Configuration Identification and Management	W	A	A	A	W	A	NA	A	A			
Configuration Audits	A	A	S	NA	A	A	NA	A	W			
Configuration Control and Status Accounting	A	A	A	NA	A	W	NA	W	A			
Configuration Management Library and Tools	W	NA	W	S	S	W	A	NA	W			

Merge Individual Ratings into a Single Capability Assessment Rating

Version 1.0 S - Strong A - Acceptable W - Weak H - High M - Moderate L - Low NA - Not Applicable NE - Not Evaluated Page ____ of ____ Pages

Figure 4-14. Example Capability Evaluation Matrix Showing Considerations that must be Merged into a Single Capability Assessment Rating

CCA SCORE SHEET

OFFEROR A

FA: Quality Management and Product Control

CCA: Software Configuration Management

GENERAL COMMENTS ON CRITICAL CAPABILITY AREA (Strengths, Weaknesses, and Risks)

Lack of established CM organization. Software CM is not well integrated into the systems engineering process. Software patches not addressed in SDP or CM plan, no prohibition against using patches.

CRITICAL CAPABILITIES:	CAPABILITIY ASSESSMENT	RISK ASSESSMENT	SPECIFIC COMMENTS ON CRITICAL CAPABILITIES (Strengths, Weaknesses, and Risks)
SCM Planning	S	L	Track record of good CM plans. Commitment to CM plans in SEMP and SEMS. Common team standard for CM plans to ensure a consistent team approach.
Baseline/Configuration Identification and Management	W	H	Lack of a consistent process for establishing and controlling baselines. No developmental CM for software design and test. Incremental builds are not addressed.
Configuration Audits	A	L	Independent group within the CM organization dedicated to audits. Audit process and personnel not well integrated with the rest of CM.
Configuration Control and Status Accounting	W	H	Lack of documented procedures for handling change requests and problem reports. Little historical evidence that this capability has been successfully applied.
Configuration Management Library and Tools	A	M	New software library approach is being proposed. No plan for implementing the library, no OPR identified, and risks have not been assessed.
CCA OVERALL SCORE	W	M	

S - Strong A - Acceptable W - Weak H - High M - Moderate L - Low Version 1.0 Page ____ of ____ Pages

Figure 4-15. Example CCA Score Sheet

CAPABILITY EVALUATION MATRIX

OFFEROR A

FA Quality Management and Product Control

CCA Software Configuration Management

CRITICAL CAPABILITIES:	EVALUATION CONSIDERATIONS: Compatibility of Approach Among Team Members and Primes/Subs is Demonstrated	Proposed Approach is Consistent With the Other Volumes of the Proposal	Responsibility for the Proposed Approach is Identified	Proposed Approach is Documented in Contractual Vehicles (SDP, SEMP, SEMS)	Proposed Approach Meets the SDCE Model Criteria	VALIDATION CONSIDERATIONS: Proposed Approach is Adequately Defined and Documented	For New Capabilities, an Adequate Analysis of Benefits vs Risks is Provided	Samples are Relevant and Demonstrate the Application of the Capability	This Capability has Been Integrated With Other Proposed Capabilities	RISK CONSIDERATIONS: Probability of Failure for the Proposed Approach	Potential Impact of Failure for the Proposed Approach	Correctability of the Deficiency of the Proposed Approach
SCM Planning										L	H	L
Baseline/Configuration Identification and Management										H	H	M
Configuration Audits										L	L	L
Configuration Control and Status Accounting										M	H	M
Configuration Management Library and Tools										L	M	M

Merge Individual Ratings into a Single Risk Assessment Rating

Version 1.0 S - Strong A - Acceptable W - Weak H - High M - Moderate L - Low NA - Not Applicable NE - Not Evaluated Page_____ of _____ Pages

Figure 4-16. Example Capability Evaluation Matrix Showing Considerations that must be Merged into a Single Risk Assessment Rating

established, the SDCE team records it in the "Risk Assessment" column on the CCA Score Sheet (see figure 4-15). In addition, short descriptions of the moderate and high risk items should be recorded in the "Comments" section of the CCA Score Sheet. Again, its very important for the SDCE team to capture descriptions of the moderate and high risk items in order to support the narrative write-up of the SDCE findings.

4.I.1.5 Prepare and Release Additional CRs and DRs

Deficiencies identified while developing final findings should be documented in a DR. Identified deficiencies are derived only from the evaluation of each offeror's proposal against evaluation standards and assessment criteria, and then, only when the proposal fails to meet the government's minimum level of compliance. Deficiencies must not be derived from a comparative evaluation of the relative strengths and weaknesses of competing offerors' proposals.

If additional clarification is needed or if action is needed to document an offeror's proposal in a contractual vehicle, then a CR should be generated. When data provided in the proposal or at the site visit is inadequate or if contradictory statements are found, a CR should be developed.

4.I.2 Score SDCE Results

The SDCE scoring method is structured to meet the requirement of the source selection process to determine the strengths, weaknesses, and risks associated with each proposal. The scoring system must, at a minimum, include color codes and written narratives at the Factor and Subfactor levels. The objective of the scoring method is to display an assessment of all the important aspects of the offeror’s proposed software development capability. If discussions are opened, the proposals are normally rated twice. The initial rating is given upon completion of the evaluation of the initial proposal; this rating is revised at the end of discussions after BAFOs are received.

A fundamental part of the standard source selection scoring process is that significant weaknesses and risks noted at lower levels must be propagate upward into the next higher level of the source selection structure. Therefore, care must be taken to ensure that significant findings at the lower levels are not lost when combining results for roll-up to higher levels. Conversely, care must also be taken to ensure that only significant findings relative to the whole project are identified for roll-up to higher levels.

4.I.2.1 Roll up CC Results to CCA Level

CCA scores are developed from a roll-up of the CC capability and risk assessments that are documented on the CCA Score Sheet (see figure 4-15). The "Specific Comments on CCs" section of the score sheet describes the strengths, weaknesses, and risks associated with each CC. In addition, the “General Comments on CCA” section describes strengths, weaknesses, and risk that are common to the entire CCA or that apply to multiple CCs. At the bottom of the score sheet there is a block to assign an overall capability and risk assessment score to the CCA.

To score the CCA, the SDCE team must combine the individual CC ratings into an overall score. The SDCE team should take into account the specific nature of the strengths, weaknesses, and risks as described in the “Comments” section. In addition, the criticality of the individual CCs to the proposed program should also be considered. General guidelines for developing the CCA capability and risk scores are shown in tables 4-18 and 4-19.

Table 4-18. CCA Capability Roll-up Guidelines

Rating	Description
Strong	The majority of the CCs are strong and there are no significant weak CCs.
Acceptable	The majority of the CCs are acceptable and there are no significant weak CCs.
Weak	There are significant weak CCs.

When an overall score for the CCA is determined by the SDCE team, it is documented in the “CCA Overall Score” block at the bottom of the score sheet.

Table 4-19. CCA Risk Roll-up Guidelines

Rating	Description
Low	The majority of the CCs are low risk and there are no significant high risk CCs.
Moderate	The majority of the CCs are moderate risk and there are no significant high risk CCs.
High	There are significant high risk CCs.

4.I.2.2 Roll up CCA Results to Functional Area Level

Functional Area scores are derived in a similar manner to the CCA scores. FA scores are determined by a roll-up of the CCA capability and risk assessments. The roll-up is accomplished on the FA Score Sheet (see figure 4-17), which is similar to the CCA Score Sheet. The first step of the FA scoring process is to transfer the overall CCA scores from the CCA Score Sheets to the “Capability Assessment” and “Risk Assessment” columns on the FA Score Sheet. The second step is to transfer the comments on strengths, weaknesses, and risks. The comments cannot normally be directly transferred. Because multiple CCAs exist for each FA, the strength, weakness, and risk descriptions may need to be merged or abstracted to a higher level. In addition, the SDCE team should select only the highest priority items to move forward to the FA level. The third step of the FA scoring process is to combine the individual CCA capability and risk scores into an overall score for the FA. The SDCE team must take into account the specific nature of the strengths, weaknesses, and risks as described in the “Comments” section of the score sheet. In addition, the criticality of the individual CCAs to the proposed program should be considered. General guidelines for developing the FA capability and risk scores are shown in tables 4-20 and 4-21.

Table 4-20. FA Capability Roll-up Guidelines

Rating	Description
Strong	The majority of the CCAs are strong and there are no significant weak CCAs.
Acceptable	The majority of the CCAs are acceptable and there are no significant weak CCAs.
Weak	There are significant weak CCAs.

Table 4-21. FA Risk Roll-up Guidelines

Rating	Description
Low	The majority of the CCAs are low risk and there are no significant high risk CCAs.
Moderate	The majority of the CCAs are moderate risk and there are no significant high risk CCAs.
High	There are significant high risk CCAs.

FA SCORE SHEET

OFFEROR A

FA Quality Management and Product Control

GENERAL COMMENTS ON FUNCTIONAL AREA (Strengths, Weaknesses, and Risks)
Lack of emphasis on defining and measuring the attributes of quality software. Minimal coding standards. Untried CM library and tools.

CRITICAL CAPABILITY AREAS:	CAPABILITIY ASSESSMENT	RISK ASSESSMENT	SPECIFIC COMMENTS ON CRITICAL CAPABILITY AREAS (Strengths, Weaknesses, and Risks)
Software Quality Management	A	M	Documented company SQA policy. Comprehensive SQA plan. Software quality not quantitatively defined. Lack of procedures for documenting and tracking discrepancies.
Software Quality Assurance	S	L	SQA performed by a strong, independent QA organization. Defined process for auditing software products. Lack of audits for verifying process compliance.
Defect Control	W	M	Lack of a process for controlling defects. Data on defects is not systematically collected nor analyzed. No evidence of defect prevention on past programs.
Metrics	A	L	Defined set of management indicators to measure progress. Metrics established for software size control, I/O utilization, and throughput. Few process metrics.
Peer Reviews	S	L	Well defined and practiced peer review process. Procedures for conducting peer reviews and the required participants and responsibilities are documented.
Internal Independent Verification and Validation (IIV&V)	A	L	IIV&V approach well defined. Software elements requiring IV&V identified. Lack of defined criteria for selecting software that requires IV&V.
Software Configuration Management	W	M	Track record of good CM plans. Lack of documented procedures for handling change request and problem reports. Risks and plans for CM library not addressed.
Documentation	A	M	Lack of standards for developing and managing software documentation. Lack of procedures for ensuring software documentation is kept current and consistent.
FA OVERALL SCORE	A	M	

S - Strong
A - Acceptable
W - Weak

H - High
M - Moderate
L - Low

Version 1.0

Figure 4-17. Example FA Score Sheet

When the single score for the FA is established by the SDCE team, it is recorded in the "Overall Score" block at the bottom of the FA Score Sheet. The SDCE team also describes strengths, weaknesses, and risks that are common to the FA in the "General Comments" section of the score sheet.

4.I.2.3 Roll up FA Results to SDCE Level

The final level of roll-up establishes the top-level ratings and strengths, weaknesses, and risks for the entire SDCE. SDCE scores are determined by a roll-up of the FA capability and risk assessments. The roll-up is accomplished on the SDCE Score Sheet (see figure 4-18), which is similar to the FA Score Sheet. The first step of the SDCE scoring process is to transfer the overall FA scores from the FA Score Sheets to the "Capability Assessment" and "Risk Assessment" columns on the SDCE Score Sheet. The second step involves transferring the comments on strengths, weaknesses, and risks. The SDCE team should select only the highest priority items to move forward to the SDCE level. Normally this will require merging or abstracting the findings to a higher level. The third step of the SDCE scoring process is to combine the individual FA capability and risk scores into an overall score for the SDCE. Figure 4-19 shows a macroscopic view of the entire roll-up process.

SDCE SCORE SHEET

OFFEROR A

GENERAL COMMENTS ON SOFTWARE DEVELOPMENT CAPABILITY (Strengths, Weaknesses, and Risks)
Well defined software development process and standards Lack of domain knowledge. Lack of interfaces between the systems, hardware, and software groups. Risk identification and management efforts are incomplete. Lack of attention to supportability and maintainability.

FUNCTIONAL AREAS:	CAPABILITIY ASSESSMENT	RISK ASSESSMENT	SPECIFIC COMMENTS ON FUNCTIONAL AREAS (Strengths, Weaknesses, and Risks)
PROGRAM MANAGEMENT	A	L	Detailed planning procedures in place. Strong statusing system. Ability to produce good work packages not demonstrated. Subcontract management is poor.
SYSTEMS ENGINEERING	W	M	Robust requirements process. Critical dependencies between software and hardware not identified. Inadequate plans for ensuring facilities and tools are in place.
SOFTWARE ENGINEERING	A	L	Documented estimating process. Excellent SDP. No formal method for managing changes to software requirements. Minimal software coding standards.
QUALITY MANAGEMENT AND PRODUCT CONTROL	A	M	Comprehensive SQA plan. Strong SQA organization. Well established peer reviews. Lack of procedures for tracking deficeiencies. Lack of a process for controlling defects.
ORGANIZATIONAL RESOURCES AND PROGRAM SUPPORT	W	M	Well defined and practiced software development process Critical training needs not addressed. Lack of a software staffing approach. Little process improvement.
PROGRAM SPECIFIC TECHNOLOGIES	A	L	AI development process integrated into the overall software development process. Strong experience base in AI. No test strategy for AI.
SDCE OVERALL SCORE	Y	M	

B - Blue
G - Green
Y - Yellow
R - Red

S - Strong
A - Acceptable
W - Weak

H - High
M - Moderate
L - Low

Version 1.0

Figure 4-18. Sample SDCE Score Sheet

4.I.3 Develop Source Selection Inputs

For direct incorporation into the source selection evaluation, the SDCE results must be compared against evaluation standards, assigned color codes, and assessed for risk. Source selection also requires that narrative assessments be written at the Factor summary level and may include lower levels as necessary. Each Factor assessment must be precise, identify the color rating, proposal risk and performance risk assessments (performance risk is determined by the PRAG). The narrative also highlights significant strengths and weaknesses.

4.I.3.1 Compare Results to Evaluation Standards

An evaluation standard establishes a uniform baseline against which an offeror’s proposal is compared to determine its value to the government. Evaluation standards are prepared for each specific criterion (Area, Factor, Subfactor). A standard may be either quantitative or qualitative, depending upon the Factor or Subfactor it addresses. The evaluation standard establishes the minimum level which an offeror’s proposal must meet in any Factor or Subfactor to be judged acceptable (green). The SDCE team only uses the evaluation standard to develop the color rating for the SDCE. The offerors are compared to one another by the Source Selection Advisory Council (SSAC). Examples of evaluation standards are shown in Volume 2, attachment 2-2.

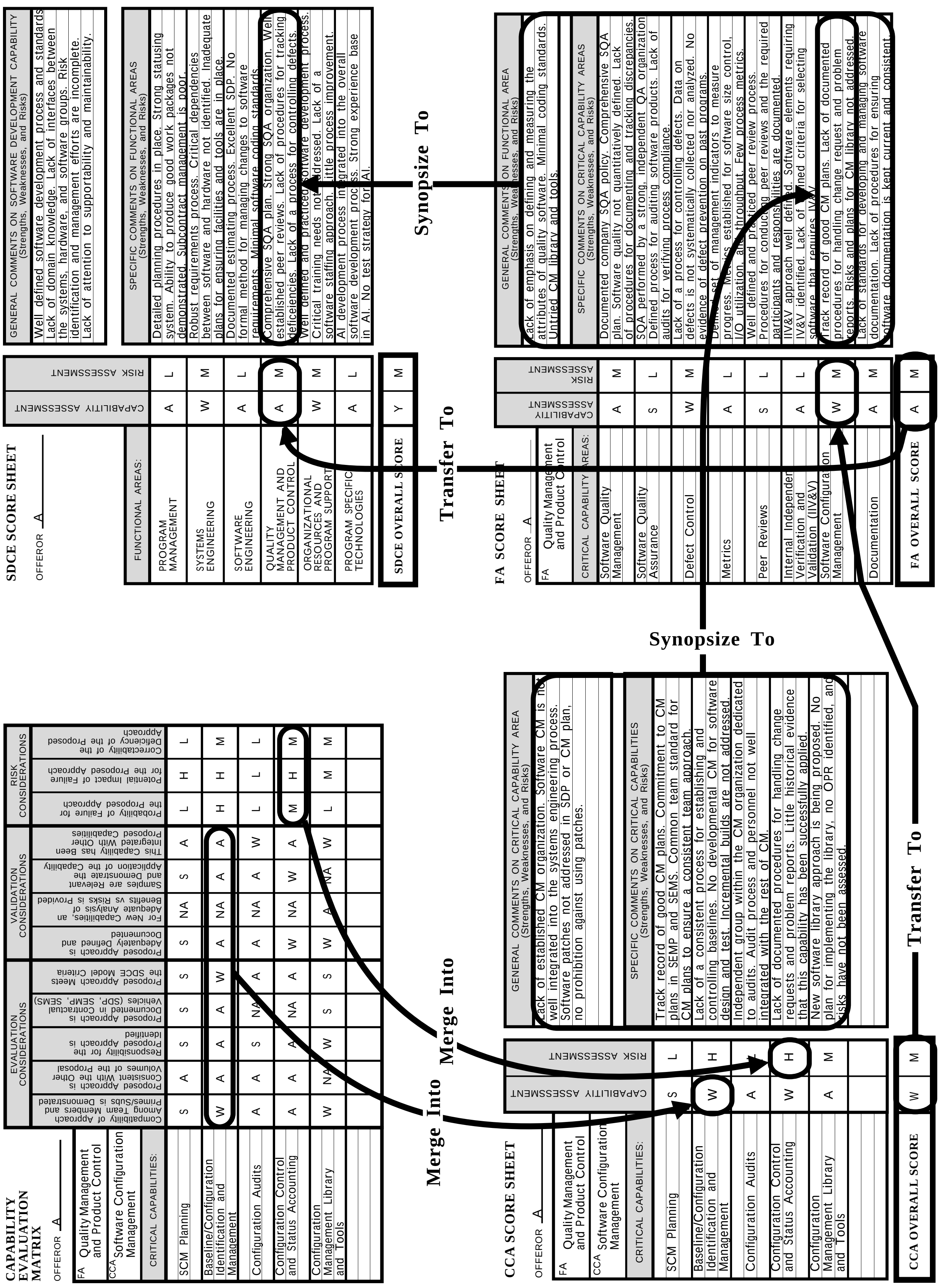


Figure 4-19. SDCE Roll-up Process

The evaluation standards for the SDCE are written so they tie back to the CCs, CCAs, and FAs. How well the evaluation standard is met is determined by examining the results of the capability assessment ratings documented on the SDCE and FA Score Sheets, as appropriate.

4.I.3.2 Assign Color Codes

Color codes are used in the source selection scoring process as a visual indicator of the adequacy of an offeror's proposal. Color ratings are mandatory at the Factor and Subfactor levels; they may be applied at the Element level, although symbols may be used as an alternative. The color rating depicts how well the offeror's proposal meets the evaluation standards. If the standard is met, the offeror is scored as green. If the requirement is exceeded, the offeror is scored as blue. If an offeror's proposal is evaluated as unacceptable (red) at any level (Factor, Subfactor, Element), this fact must be included in the rating and the narrative assessment at that level and each higher level. A positive proposal presentation is not listed as a strength and does not receive an exceptional rating unless the offeror's performance is assured via contractual incorporation of the strength or by evidence in the proposal that the contractor's current business practices or corporate structure will yield the desired result.

After the SDCE team has compared the findings to the evaluation standards it assigns each Factor, Subfactor, and Element (if applicable) a color code. Table 4-22 defines the color codes from AFFARS AA.

Table 4-22. Source Selection Color Codes

Color	Rating	Definition
Blue	Exceptional	"Exceeds specified performance or capability in a beneficial way to the government, and has no significant weaknesses."
Green	Acceptable	"Meets evaluation standards, and any weaknesses are readily corrected."
Yellow	Marginal	"Fails to meet evaluation standards, however any significant deficiencies are correctable."
Red	Unacceptable	"Fails to meet a minimum requirement of the RFP, and the deficiency is uncorrectable without a major revision of the proposal."

4.I.3.3 Assign Risk Ratings

Along with each color code, the SDCE team must assign a risk rating that reflects the software development capability risk associated with the offeror's proposal. The risk and color ratings assigned to any Factor or Subfactor are independent of each other and have equal weight. Any risk assessment rating may be used with any color rating as necessary to reflect the results of the SDCE. Table 4-23 defines the risk ratings from AFFARS AA.

Table 4-23. Proposal Risk Ratings

Symbol	Rating	Definition
L	Low	"Has little potential to cause disruption of schedule, increase in cost, or degradation of performance. Normal contractor effort and normal government monitoring will probably be able to overcome difficulties."
M	Moderate	"Can potentially cause some disruption of schedule, increase in cost, or degradation of performance. However, special contractor emphasis and close government monitoring will probably be able to overcome difficulties."
H	High	"Likely to cause significant serious disruption of schedule, increase in cost, or degradation of performance even with special contractor emphasis and close government monitoring."

4.I.3.4 Write up Factor and Area Summaries

Source selection guidance does not require the use of Factor or Area summaries; they are, however, widely used by source selection teams to document the narrative assessment. In addition, no standard format exist for the summaries. Consequently, the SDCE team will have to adjust the write-up of their findings to comply with the local Factor and Area summary formats. Summaries should indicate, as a minimum, what is offered, whether it meets or fails to meet the standard, any strengths or weaknesses, what may be done to remedy a deficiency, the impact of any deficiency, and a risk assessment of the offeror's proposal approach and ability to perform. Figure 4-20 is an example of a factor summary.

4.I.3.5 Provide SDCE Input to SSEB Executive Report and Briefing

After the evaluation teams have completed their evaluation of the BAFOs, the SSEB chairperson compiles and presents the SSEB's overall evaluation results to the SSAC in two forms, a written executive summary report and an oral presentation.

The report and presentation should include ratings (both color and proposal risk) and narrative assessments (which identify strengths and weaknesses and support color and proposal ratings). The objective of the SSEB executive summary report is to present an evaluation of each proposal against solicitation requirements based on established evaluation criteria and standards. An audit trail from the highest to lowest elements of the evaluation must be provided by supporting documentation. The SDCE team should assist the SSEB chairperson in preparing the SSEB executive summary report relative to the SDCE findings. A typical part of the executive presentation is the matrix evaluation chart, which shows at a glance the color codes and risk ratings for an Area or Factor. Figure 4-21 is an example matrix evaluation chart with the SDCE listed as a Factor.

Factor Summary

Area: Technical	**Factor:** Software Development Capability	**Offeror:** ABC Corporation	**Color Rating:** Yellow

Description of Proposal:

The sumary findings for this offeror are:

STRONG

- None

ACCEPTABLE

- Program Management
- Software Engineering
- Quality Management and Product Control
- Program Specific Technologies

WEAK

- Systems Engineering
- Organizational Resources and Program Support

Strengths and Weaknesses:

STRENGTHS

- Well defined development processes and standards
- Strong experience base in applying proposed approach on other programs
- Software development process documented in Software Development Plan

WEAKNESSES

- Lack of structured process for defining and controlling baselines
- No established mechanism for ensuring testing deficiencies are tracked to closure

(see additional strengths and weaknesses on back)

Risk Assessment:

MODERATE RISK: The offeror's software development process is well documented and institutionalized within the company. However, critical dependencies between hardware and software are not identified. There are inadequate plans for ensuring facilities and tools will be in place. The baselining and defect tracking processes are weak.

Factor Chief Signature:	**Area Chief Signature:**

Figure 4-20. Example Factor Standard

MATRIX EVALUATION

Offeror: ABC

Area: Technical

Factors

	Reliability Maintain-ability and Producibility	Airframe Design	Software Development Capability	Flight Control System Design	Navigation, Comm, and Displays
Proposal Rating	G	G	Y	B	G
Proposal Risk	L	M	M	M	L
Performance Risk	M	M	M	L	L

Figure 4-21. Example Matrix Evaluation Chart

4.I.4 Integrate SDCE Results into Source Selection

The SDCE team needs to coordinate its findings with other source selection teams. The coordination of findings between the various teams is important in developing an integrated and consistent assessment of the offeror's proposal.

4.I.4.1 Determine Mapping of SDCE CCAs into Collateral Factors and Areas

To integrate the SDCE results into the rest of the source selection, the SDCE team must first identify the other Areas, Factors, and Subfactors that correspond to SDCE CCAs. Figure 4-22 shows an example breakdown of Areas, Factors, and Subfactors; the solid boxes indicate a Factor or Subfactor that correlates to one or more CCAs. Notice that the SDCE results may be applicable in various other areas of the source selection and may also occur at different levels.

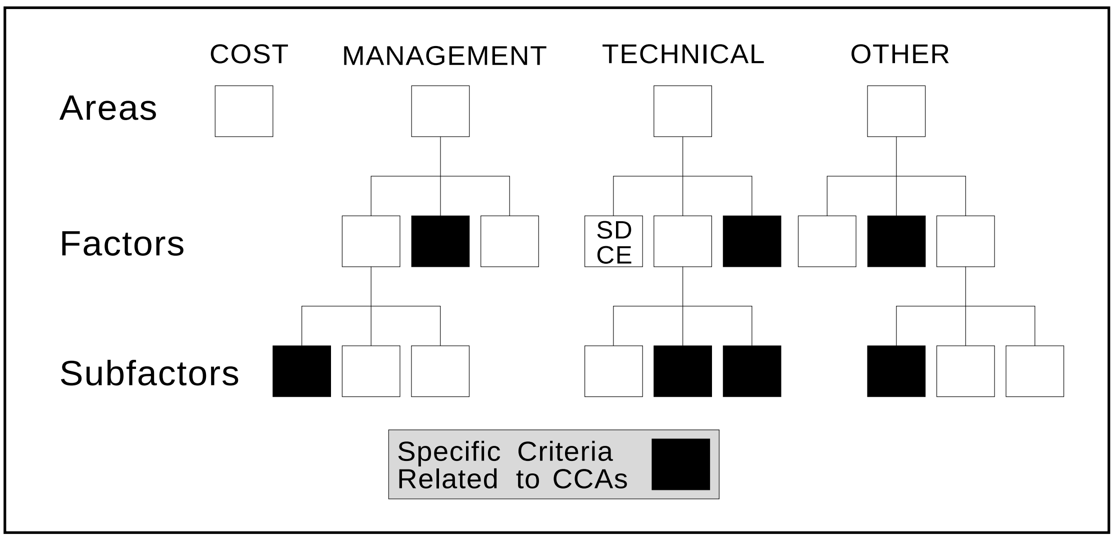


Figure 4-22. Example Mapping of CCAs into Other Specific Criteria

4.I.4.2 Work with Factor and Area OPRs to Integrate SDCE Findings

Once the other relevant Areas, Factors, and Subfactors are identified, the SDCE team needs to work with those teams to ensure an integrated and consistent picture is developed. In addition, the SDCE team should coordinate their results with the PRAG (if used).

4.I.4.3 Assist Factor and Area OPRs in Preparing Factor and Area Summaries

The SDCE team should provide narrative input as needed to the other relevant Area, Factor, and Subfactor teams. The other relevant Area and Factor summaries should also be reviewed by the SDCE team to ensure consistency with the software development capability findings.

4.I.4.4 Develop and Deliver SDCE Findings Briefing to SSEB

To assist in the evaluation of the proposals, the SDCE team should develop and deliver a briefing to the rest of the SSEB to inform them about the results of the offeror's software development capability. An example format for this SDCE briefing is shown in Volume 2, attachment 2-9.

4.I.5 Assist in Evaluating Offerors

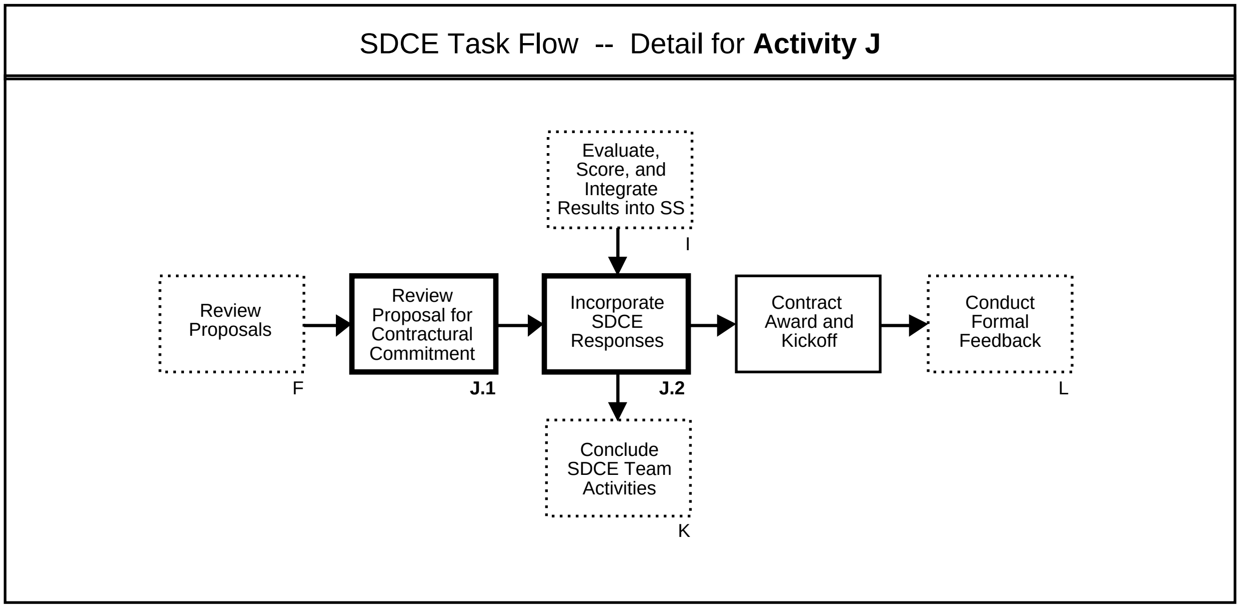
4.I.5.1 Assist SSEB in Preparing and Delivering Formal Presentation to SSAC or SSA

The SSEB chairperson is responsible for briefing the results of the source selection evaluation to the SSAC. However, the chairperson does not have to personally present the entire briefing. The SDCE team should assist the SSEB chairperson in developing the part of the presentation dealing with the results of the SDCE and should be ready to brief the SDCE results if asked.

4.I.5.2 Consult with SSAC as Needed

It is the responsibility of the SSAC to compare the offerors' proposals to each other. This comparison is based on an analysis of the evaluation performed by the SSEB. The SDCE team should be ready to assist the SSAC with any questions or concerns about the results of the SDCE evaluation or how it was conducted.

Section 4.J Incorporate into Contract



One of the significant advantages of performing an SDCE as part of the source selection process is the opportunity it provides to gain contractual commitment from the offerors to follow their proposed engineering and related development processes. The SDCE explicitly solicits the offeror to define its capability in terms of systems and software engineering and directly related development processes. This section provides guidance to the SDCE team and other SSEB participants in the following:

- *How to use the SDCE method to gain contractual commitment to process*

- *How to work within the SSEB to incorporate offerors' capabilities into the SDP, SEMP, and SEMS*

4.J.1 Review Proposal for Contractual Commitment

An offeror's SDCE proposal information describes processes it plans to apply on the subject program. This information includes responses to model questions concerning specific engineering and development processes, internal company standards and operating instructions that define company processes, and examples from programs on which these processes have already been applied.

The key proposal documents for gaining contractual commitment to the development processes are the SDP, SEMP, and SEMS. The proposal-level SDP is a draft plan that should define and describe the software development processes to be used in the development of software on the contract. Similarly, the draft SEMP describes the engineering processes to be applied on the contract. The SEMS defines events with completion criteria for the processes identified in the SDP and SEMP.

This event-driven development approach is directed by DoD 5000 series documents. Like the model contract, the SEMS is subject to modification during the source selection period and serves as a model or agreed-to starting baseline prior to contract award. These documents are carried through program execution after program award. (Note: Other documents, such as an Integrated Management Plan or Integrated Master Schedule, may serve the same purpose of defining the development process in terms of events with completion criteria.)

The key point is to carefully review the SDP, SEMP, and SEMS to verify that they are correct, adequate, complete, and consistent with the SDCE information gathered through the proposals and site visits. The processes, events, and completion criteria in the SDP, SEMP, and SEMS should be sufficiently well defined to ensure that they incorporate key software development processes essential to successful development.

Significant discrepancies should be identified and addressed in CRs and noted for discussion during the SDCE site visits. Also, all noted discrepancies should be identified, recorded, and held for resolution with the offeror after contract award.

The information in the proposal solicited by the SDCE should also be reviewed for responsiveness to and consistency with systems and software engineering process-based award fees.

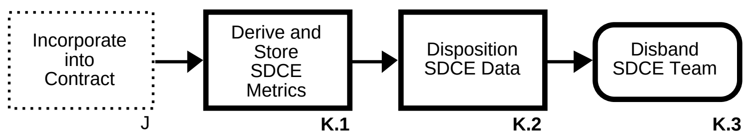
4.J.2 Incorporate SDCE Responses

When an SDCE question elicits a contractor response that defines adequate engineering processes intended to be applied on the subject program, that response must be incorporated into the SDP, SEMP, and SEMS that are placed on contract. This is important because the responses to the SDCE questions are not contractually binding. If, for example, the site visit clarifies a significant process approach, the SDCE team should prepare a CR to request that this clarification be incorporated into the SDP, SEMP, and SEMS.

The incorporation of software development processes from both the SDCE proposal information and the site visit dialogue will contribute significantly to establishing a baseline definition and description of the offeror's fundamental software and related systems development processes.

Section 4.K Conclude SDCE Team Activities

SDCE Task Flow -- Detail for **Activity K**



This chapter describes the concluding SDCE team tasks. These tasks include capturing metrics data, preparing lessons learned reports, ensuring correct handling and disposition of the data collected during the application of the SDCE, and formally disbanding the team. Guidance will be provided for the following:

- *How to decide what metrics data should be captured*
- *Where to forward the metrics data*
- *How to disposition the SDCE data*
- *When to disband the SDCE team*

4.K.1 Derive and Store SDCE Metrics

Throughout the SDCE process, the team needs to document:

- Resources and effort required to perform the SDCE
- Information that will become part of the formal feedback to offerors
- Offeror comments on the conduct and value of the SDCE
- Recommended improvements to the method
- Lessons learned in performing the SDCE

Lessons learned in performing the SDCE would include:

- New SDCE model criteria and questions used, i.e., tailored for the particular program
- Clarification of model questions considered to be ambiguous
- Problems with the SDCE method, model, or process
- Strengths or weaknesses of the method

Two distinct types of metrics need to be collected from the individual application of the method during source selection: those that will be used in improving the SDCE method and those that identify specific contributions to the selection process and follow-on program execution. While the metrics dealing with improving the method are relatively straightforward, it is a challenge to identify real contributions to the selection process and follow-on program executions. Recognizing this difficulty, the metrics to be collected are organized in two sets, one called “SDCE Method Improvement Metrics,” the other “Source Selection and Program Execution.”

To support the recording of these metrics, two forms are provided in Volume 2, attachment 3-9 that can be used as metrics collection record sheets. To help interpret the metrics, an example of filled-out forms is provided in Volume 2, attachment 2-10. These templates are based on the same hypothetical program as in the SDCE Implementation Plan, Volume 2, attachment 2-1.

4.K.2 Disposition SDCE Data

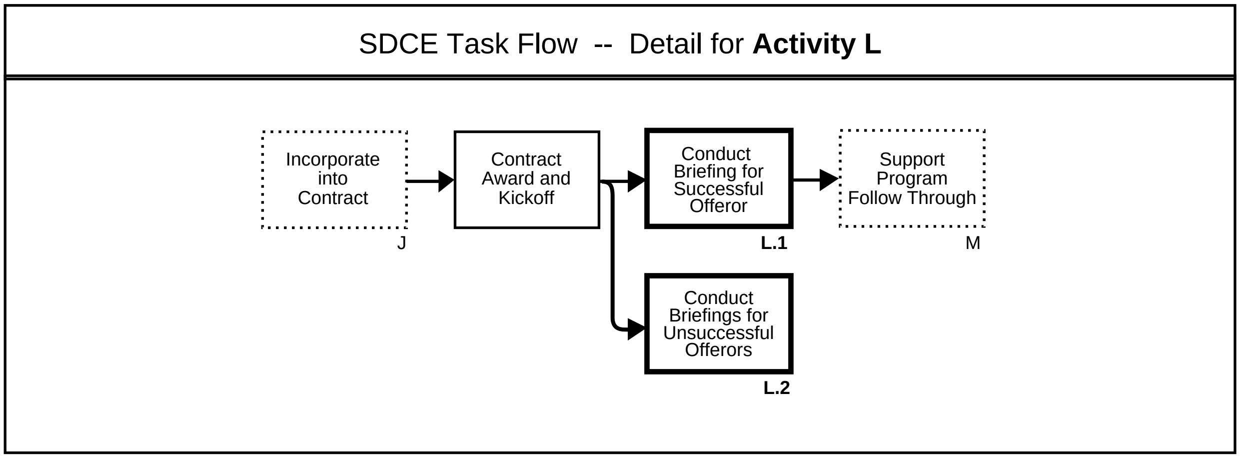
All data gathered as part of the SDCE process must be properly dispositioned. Since the SDCE is an integral part of the source selection process, all data, including site visit data, must be handled and dispositioned in accordance with source selection guidance in AFFARS, Appendix AA. Data associated with the contractors who are awarded contracts may be used by the program office in conjunction with the acquisition development effort of those programs.

Metrics information relating to the conduct and experience of performing the SDCE should be forwarded to the local Center SDCE OPR, who will forward appropriate metrics data to the AFMC SDCE OPR. This metric data will be used to improve the SDCE method.

4.K.3 Disband SDCE Team

Upon completing the final evaluation writeups of the offerors’ SDCE information and recording the metrics, the SDCE team can be disbanded. Each member will be outbriefed by the SSEB in accordance with established source selection procedures. The program stakeholder members of the SDCE team will continue to support the SSEB chairperson and the follow-on post-award contractor feedback activities. Additional post-award activities may include supporting the prime contractor in using parts of the SDCE method (minus the government source selection specific material) to select subcontractors. Also, the leader of the SDCE team, if not a program stakeholder, may be asked to continue to support the SSEB chairperson through the feedback activities.

Section 4.L Conduct Formal Feedback



It is in the best interest of the government to foster continuous improvements in the capabilities of offerors. To accomplish this, the offerors need detailed feedback on the results of the SDCE. After contract award, SDCE feedback sessions must be made available to all offerors. The intent is to provide an atmosphere of open communication which allows comprehensive software development capability evaluation data to be shared between the evaluation team and each offeror within the legal constraints of the FARs. This data is intended to assist all offerors, successful and not, in planning further process improvements as a result of the lessons learned from the SDCE process. This section covers the following:

- *How to prepare and conduct formal feedback sessions for the successful offeror*
- *How to prepare and conduct formal feedback sessions for the unsuccessful offerors*

4.L.1 Conduct Formal Feedback Briefing for Successful Offeror

The feedback session to the successful team (prime contractor and team members) can help build a strong acquisition/development team atmosphere that will extend through the life of the contract, based on a mutual understanding of the contractor's capability and processes at contract start. The feedback session should be attended by the contractor software personnel assigned to the program who can commit the organization to improvements.

The feedback session is conducted by the SSEB, program office, and former SDCE core team members at a mutually agreed to place and time. Generally, the feedback session is held at the prime contractor's facilities. It consists of viewgraph presentations and working meetings; therefore, multiple meeting rooms with viewgraph machines need to be reserved. The contractor is encouraged to submit a list of items for discussion at the time the meeting is arranged.

The facility should accommodate the SSEB and SPO personnel, prime contractor and teaming partners, and any invited subcontractor personnel. The contracting agency will identify the number of goverment personnel and any special requirements at the time the meeting is arranged. The hosting contractor should provide maps to the facility, the phone number of the escort to be called on arrival, the name and phone number of the local DPRO representative, and, if a security clearance is required, the contact's name and phone number.

The contractor is requested to provide an evaluation of the SDCE process for presentation at the feedback session. The data to be presented should include:

- The effort involved in planning and preparing for the SDCE in person days
- The effort to support the site visit in person days
- The effort to support follow-on activities after the site visit and before award in person days
- Any feedback on the method, both positive and negative, that can be used to improve the SDCE

The SSEB should distribute an agenda of the meeting at least a week before the meeting is scheduled. A suggested agenda, with times, is shown in Volume 2, attachment 2-11. Suggested methods for collecting the data for the presentation are also provided. A sample presentation is shown in Volume 2, attachment 2-12. The presentation must be confined to data concerning the evaluation of the successful offeror.

4.L.2 Conduct Formal Feedback Briefings for Unsuccessful Offerors

The source selection process provides for feedback sessions, on request, to each unsuccessful offeror team (prime contractor and teaming partners).

The feedback session is conducted by the SSEB, including former SDCE core team members. Generally, the feedback session is held immediately after the formal face-to-face debriefing and is attended by offeror software practitioners who can understand the level of detail of the presentation and who can effect positive change in their organization. The feedback session is a viewgraph presentation at the contracting agency's source selection facility. The facility should accommodate the SSEB, the prime bidder's personnel, and any invited subcon-tractors.

The bidder is encouraged to submit written questions at the time the meeting is arranged and is requested to provide an evaluation of the SDCE process for presentation at the feedback session. The data to be presented should include:

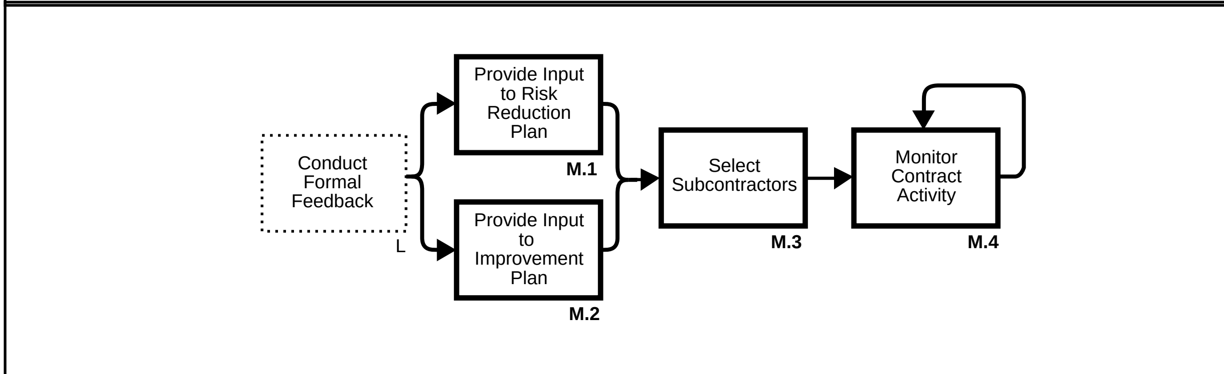
- The effort involved in planning and preparaing for the SDCE in person days
- The effort to support the site visit in person days
- The effort to support follow-on activities after the site visit and before award in person days
- Any feedback on the method, both positive and negative, that can be used to improve the SDCE

A suggested agenda, with times, is shown in Volume 2, attachment 2-13. Also, suggested methods for collecting the data for the presentation are provided. A sample presentation is shown in Volume 2, attachment 2-14. The presentation must be confined to data concerning the evaluation of the team being briefed.

After the feedback session, the data in possession of the evaluation team will be disposed of or passed to the program office as appropriate for “Source Selection Sensitive,” “Competition Sensitive,” “Proprietary,” and/or classified data, in accordance with source selection guidelines.

Section 4.M Support Program Follow-Through

SDCE Task Flow -- Detail for **Activity M**



Although the SDCE team is disbanded shortly after contract award, some members of the team will continue in their program office roles. Observations, understandings, and initial team building activities will be carried forward into actual contract execution. The SDCE results provide a rich source of information for other program management activities. This section covers specific areas that are direct follow-up to the SDCE process, including:

- *How to select subcontractors*
- *How to monitor risk management and process improvement activities*
- *How to monitor the processes and resources the contractor has committed to in the SDP, SEMP, and SEMS*
- *How to determine the completion of activities that qualify for incentive awards*

4.M.1 Provide Input to Risk Reduction Plan

Data from the risk considerations columns of the Capability Evaluation Matrix (Volume 2, attachment 3-4) should be summarized into risks that may have a significant effect on the program's success and then documented in the risk reduction plan. The selection of the risks to be carried forward into the program should be based on the methodology detailed in AFMC Risk Management Guide. Risks determined after contract award (e.g., when subcontractors are selected after contract award) should be integrated into the risk reduction plan of the affected subcontractor and may be of a serious enough nature to be included into the prime contractor's risk reduction plan.

4.M.2 Provide Input to Improvement Plan

The weaknesses described in the CCA Score Sheet and the FA Score Sheet (Volume 2, attachments 3-5 and 3-6) should be used as the basis for the improvement plan.

Improvement plans should be developed for subcontractors selected after contract award. Particular attention should be paid to processes that are common between the prime contractor and subcontractors and that therefore need a coordinated improvement plan. Likewise, progress against these plans needs to be monitored by the prime contractor and, as appropriate, the contracting agency.

4.M.3 Select Subcontractors

When a prime contract is awarded before subcontractors are selected, the SDCE method (minus the government source selection specific material) should be applied after contract award as part of the subcontractor selection process. The prime contractor, with SPO support as appropriate, should tailor the model criteria and adapt the process described in this pamphlet to the needs of the project. To support team building and to emphasize mutual commitment, members of the prime contractor's project team should evaluate the subcontractors, rather than delegating the task to another internal organization.

4.M.4 Monitor Contract Activity

The improvement plan documents the initial conditions of the contractor's capability and describes an approach for developing beneficial improvements and monitoring progress throughout the life of the contract. During the performance of the contract, the processes committed to in the SDP, SEMP, and SEMS should be monitored for compliance. As the contract evolves, the SDP, SEMP, and SEMS will need to be updated. The baseline established in the improvement plan will help ensure that commitments made in the original SDP, SEMP, and SEMS are not violated in the course of Continued Process Improvements.

As a result of the SDCE, Continuous Process Improvement activities may have been included in activities which qualify for incentive awards. If the contract has provisions for incentive awards, the milestones defining the completion of the award activities must be monitored.

CHAPTER 5. MODEL CRITERIA AND QUESTIONS

This chapter contains a complete, tabulated list of all the model criteria and questions. Each criterion is cross-indexed to the questions that support it, and each question is cross-indexed to its related criteria. This cross-indexing supports model tailoring as well as the formulation of the SDCE results. The material is organized by the six Functional Areas described in paragraph 2.1.3 and further broken down into each FA's Critical Capability Areas and Critical Capabilities (reference chapter 3).

Figure 5-1 explains the layout of a typical page.

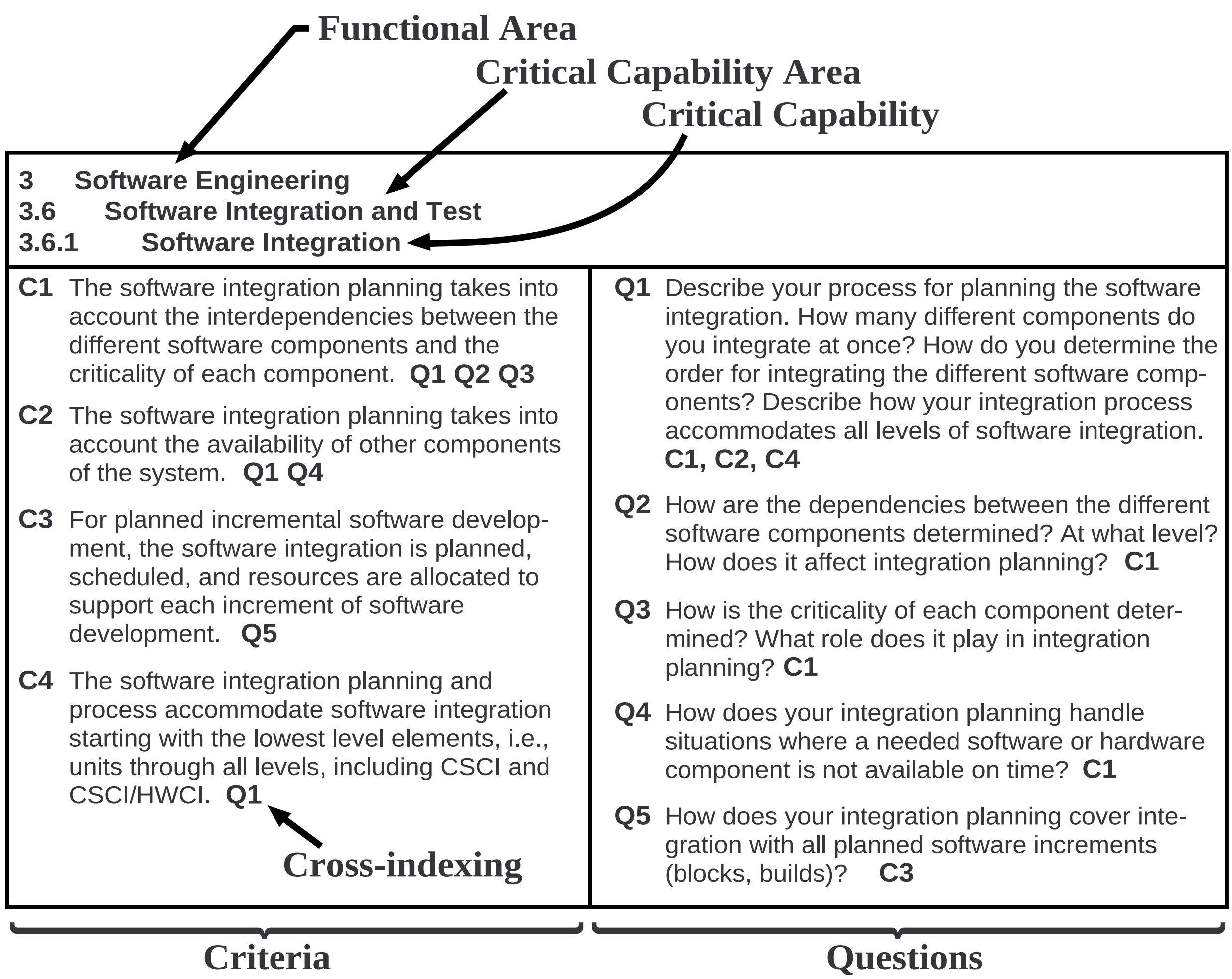


Figure 5-1. Example Format for Criteria and Questions

1 Program Management
1.1 Management Authority, Responsibility, and Accountability
1.1.1 Organizational Approach

C1 The software development and management functions are organized consistent with the proposed overall system development organizational structure (e.g., straight functional, Integrated Product Teams (IPTs)) and include identified support functions to the system engineering, subcontractor development and other functional development support activities as needed. **Q1 Q3**

C2 The program organization includes specific elements composed of systems, software and hardware engineering that are responsible for the allocation of requirements and derivation of design from the System Specification to the Software and Hardware Configuration Item specifications. **Q2**

C3 The software engineering organization is structured such that all program software (including support software) development is assigned to specific organizational elements. **Q4**

C4 The total software development organization is defined and responsibilities assigned, including identified elements responsible for the management and control of subcontractor-developed and vendor-delivered software. **Q3**

C5 The organizational structure integrates special technology driven resource requirements (e.g., specialists in languages, architectures, methods, tools not assigned full time to the program) into the program organizational working structure. **Q5**

Q1 Describe the total software development organization, top to bottom, including intermediate organizational supervisory levels.How is this software development function organizationally integrated and consistent with the programs overall system development organizational structure (e.g., straight functional, IPTs, etc.)? Describe the major software subcontractors' organizations to develop software. Describe any formal agreements between team members that define specific responsibilities for development. **C1**

Q2 Describe the organizational elements responsible for the allocation of requirements and derivation of design from the System Specification through the various levels of design and requirements to the Software Requirements Specifications, Hardware Configuration Item Specifications, Interface Requirements Specification and top level design. Include teaming and/or subcontracting elements. **C2**

Q3 How does this structure provide the necessary support functions to related development functions such as systems engineering, subcontractor development management, vendor management and other functional development support activities? **C1 C4**

Q4 Describe, within the identified software development organization and structure, the responsibility assignments for all program software including support, integration and test software. **C3**

Q5 How is the organization structured to meet the program needs for specialized technology skills that are driven by program requirements such as language and architecture specialists who may not be required full-time on the program? **C5**

1 Program Management
1.1 Management Authority, Responsibility, and Accountability
1.1.2 Management Control

C1 The chief engineer/system engineer, or equivalent is organizationally responsible for all technical activities on the program.
Q1 Q2

C2 Responsibility for control of all software development resides within the program organization including, subcontracted, simulation, integration and test software.
Q3 Q4 Q5 Q6

Q1 Where does the overall technical responsibility for the program development reside? Identify program technical activities that do not report through the lead technical manager. Where does the lead technical manager (Chief Engineer) report? **C1**

Q2 Where does the program management responsibility reside? Identify any development activity outside the control of the program manager, upon which the program is dependent , e.g. reusable software, tools to be acquired, components being developed by another organization, etc. How does the program manager influence and status these components? **C1**

Q3 Where does the overall software responsibility reside? Does all software development report through a single software manager? Is all software, including that developed by teaming associates and subcontractors, support, simulation, integration and test software, included in the overall software management responsibility? If not, how is this other software managed, interfaced and integrated with the central software effort? **C2**

Q4 How is software managed within the Integrated Product Development Team approach to the system development? How is software managed/developed within subsystems and across interrelated subsystems? **C2**

Q5 Who is responsible to see that the Software Development Plans and Software Development Standards are followed?
C2

Q6 Describe the specific management functions applied to control software development. **C2**

1 Program Management
1.2 Program Planning and Tracking
1.2.1 Planning

C1 The program planning accounts for the integration of software development and management with system and hardware management. **Q1**

C2 The proposed program planning approach includes planning for personnel qualifications, quantities, skill types, need dates and required training. **Q7**

C3 The program planning includes the necessary reviews, accountability, status assessment, schedule control and reporting to manage the software related system development activities leading to the definition of the software requirements baseline. **Q2 Q3**

C4 The program planning includes a series of technical and management reviews with associated completion criteria that is used to control the development progress. **Q3**

C5 Specific management planning processes are defined to account for the proposed software development methodologies and implementation language selected. **Q4**

C6 Variance thresholds are established for critical status metrics (e.g. size, cost effort, progress, and schedule). **Q5 Q6**

Q1 How is your software development planning integrated with systems management and hardware management? **C1**

Q2 Describe your planning process used to establish the front-end software related system development activities. Describe your process to status and report these activities including specific criteria and control measures. Who is responsible to perform these front-end management activities? **C3**

Q3 Describe your technical and management reviews used to control the development progress throughout the entire development period. Define these events and corresponding criteria. How are these events incorporated into the SEMP, SEMS, SEDS, and the SDP? **C3 C4**

Q4 Describe any special management planning processes used as a result of the selected software development methodologies and programming languages. How do these processes vary from your standard software development management activities? **C5**

Q5 Identify the software tracking metrics to be used on this program. Describe your process for monitoring critical status metrics or indicators. How do you determine when management action is required? Describe the conditions that would result in management action for each established metric or indicator. **C6**

Q6 Where is this metrics monitoring process documented? **C6**

Q7 Describe how your program planning includes provisions for personnel qualifications, quantities, skill types, need dates, and required training. **C2**

1 Program Management
1.2 Program Planning and Tracking
1.2.2 Contract Work Breakdown Structure (WBS)

C1 The proposed CWBS and internal CWBS generation procedures identify software elements to levels that support software management visibility and are compatible with cost reporting and program RFP requirements. **Q1 Q2 Q3 Q4**

C2 The proposed CWBS and internal CWBS generation procedures identify how the CWBS links with and traces to the work definition system down to and including the software work packages. **Q5**

C3 The program has a mutually consistent and integrated CWBS, work definition, scheduling, and cost tracking system and is used as the basis for program status and control. **Q6**

Q1 How and at what level is software structured in the CWBS? Provide examples of your recent CWBSs that include major software development. **C1**

Q2 What are the factors and criteria for determining software level within the CWBS? **C1**

Q3 Identify your internal standards (criteria) for identifying software work within the CWBS. **C1**

Q4 At what level (of CWBS) is software reported in the cost performance report? **C1**

Q5 Describe the overall flow of work definition from the CWBS down through detailed work definition including Cost Accounts and Work Packages. Explain (illustrate) how the software work package interfaces with and is traceable to the software elements of the CWBS. **C2**

Q6 Describe how your CWBS procedures integrate with your work definition process, scheduling process, and cost tracking system. Describe how the CWBS is used to support program status and control. **C3**

1 Program Management
1.2 Program Planning and Tracking
1.2.3 Work Packages

C1 A documented process exists for defining software development work packages, including schedules and manpower allocations. This process includes rules and criteria for formulating software work packages. **Q1 Q2 Q3**

C2 The software work package is used to manage the work and is used as the basis for cost performance reporting. The cost performance reporting system includes all of the software development tasks and activities. **Q4**

C3 The software work packages include planned and actual effort expenditures. **Q1**

C4 The software work packages have completion milestones, with associated criteria scheduled consistent with the program requirements (e.g., CPR and CSCSC). **Q1 Q5**

C5 The scheduling information contained in the software work packages is consistent with the program scheduling system. **Q3 Q5**

C6 The CWBS and Work Package Definition System provides a correlation to the software structure. **Q6**

Q1 Explain your method for defining software work package including schedules and manpower allocations. Identify rules, essential elements and criteria for an acceptable software work package within your system. **C1 C3 C4**

Q2 Identify the document that fully describes the software work package method. **C1**

Q3 Describe how your Cost Account and Work Package definition process is integrated with the system and software scheduling system. Explain how the software work package is used to plan the work, i.e., plan, define and assign resources (manpower loading) and responsibility. How are the CWBS, Cost Accounts and Work Packages used to status and report progress of the program in terms of effort (cost) and schedule? **C1 C5**

Q4 Explain how the software work package is used in cost performance reporting (CPR). Explain how CPR data such as earned value is development tasks and activities included in the CPR. How do these differ for varying elements of the CWBS? How is the CPR used to establish a true indication of software cost considering the effects of general or overhead functions such as software management, configuration management, software quality assurance, internal independent verification and validation, etc.? **C2**

Q5 How do the Work Package milestones and completion criteria correlate with SEMP/SEMS milestones? Is the completion criteria and earned value within the CPR or CSCSC system related to the SEMP/SEMS completion criteria? **C4 C5**

Q6 Explain how the CWBS, cost accounts and work packages correlate with the software structure. **C6**

1 Program Management
1.2 Program Planning and Tracking
1.2.4 Schedules

C1 Software schedules are established in sufficient detail to maintain visibility and control of the development process including the establishment of any planned blocks, builds or increments.
Q1 Q5

C2 The program's software scheduling and status system and proposed schedules are consistent and integrated with SDP and the program system level schedules, including the SEMP/SEMS/SEDS (as appropriate). **Q3 Q4 Q6**

C3 The proposed schedule duration for software development, software integration and software/hardware integration are consistent with the effort to be accomplished as estimated with established estimating models and the offeror's historical data. **Q8**

C4 The lowest level software schedules include task, phase, and milestone definitions that are consistent with the software work definition packages. **Q2**

C5 A process is defined to maintain consistent software schedule information across various disciplines including engineering, management, and the Cost Performance Reporting System.
Q7

Q1 Describe your approach to establishing the software development schedules from the top system level schedule to the lowest level detail schedules. Explain how incremental (block, build) software development schedules are established. **C1**

Q2 Define the phases, tasks, and milestones used in your most detailed (lowest level) software schedules. How are the duration of various tasks and phases of software schedules determined? How do these relate to the software work definition packages? **C4**

Q3 Describe how your process to establish software schedules integrates with the programs higher level scheduling system, e.g., SEMS and SEDS. **C2**

Q4 Describe your approach to defining milestones with completion criteria in the Systems Engineering Master Schedule (SEMS) and Systems Engineering Master Plan (SEMP). How is this approach correlated with the software scheduling system including use of software milestones and criteria? **C2**

Q5 Describe your method for monitoring and statusing software development schedules. Who is responsible for this function? How is each level of schedule that addresses software used in the management process? Which level is used as the baseline to track and report status? **C1**

Q6 Describe how software schedules are defined, referenced, used and updated in the SDP. **C2**

Q7 Describe your approach to maintaining consistent software schedule information across various disciplines e.g. management, engineering, and program control (CPR/CSCSC). **C5**

Q8 Describe this program's software development, software integration and software/hardware integration schedules including time phasing and duration. Also describe the scheduling of any planned increments or blocks. How were these schedules derived? Relate the proposed schedules to the effort to be accomplished (man months), available personnel resources and your past schedule accomplishments on similar programs. **C3**

1 Program Management
1.3 Subcontractor Management
1.3.1 Capability Evaluation

Criteria	Questions
C1 As part of the subcontractor selection process, documented procedures exist to evaluate subcontractor's capability and capacity to develop software. **Q1 Q2**	**Q1** How are potential subcontractors' software development capabilities and capacities evaluated prior to selecting a specific subcontractor? **C1** **Q2** Where is this procedure for evaluating subcontractors' software capabilities and capacities documented? **C1**

1 Program Management
1.3 Subcontractor Management
1.3.2 Subcontractor Development Management

C1 The proposed subcontractor management process is integral to the system program management process and provides integrated reporting and control of the subcontractor software development activities consistent with the program's management control system. **Q1**

C2 The system-level engineering management controls including SEMP/SEMS/SEDS are levied on subcontractors. **Q2**

C3 Periodic management and technical reviews to address subcontractor development progress are conducted and are reflected in the program's SDP/SEMP/SEMS/SEDS. **Q1**

C4 A process is defined to specify and control the subcontractor's performance requirements, interfaces, deliverables and product testing. **Q3**

C5 A documented process exists which requires reviewing and assessing the technical content of subcontractor generated design information and documentation. **Q4 Q5 Q6 Q7 Q8**

C6 The software test and verification process includes subcontractor developed software and incorporates the subcontractor software test and verification management and results into the overall hierarchical test process. **Q9**

C7 The subcontractor's defined software cost status and reporting system is compatible with the program cost status and reporting requirements. **Q10 Q11**

C8 The software size control program established for the program is applied to the subcontractor effort and monitored throughout the development. **Q12**

Q1 Fully describe your process for subcontractor management including reporting and control of the subcontractor software development activities. How does this process relate to and integrate with your overall system program management approach? Describe how the subcontractor management and review activities are reflected in the program level SEMP/SEMS/SEDS. **C1 C3**

Q2 Using this approach, how are the SEMP, SEDS, SEMS, and the SDP flowed down to the subcontractors? **C2**

Q3 How do you specify and control the subcontracted software technical/performance requirements, interfaces, deliverables and product testing (test requirements and criteria)? **C4**

Q4 Describe your process for establishing and conducting periodic management and technical reviews and interchanges with your subcontractors. **C5**

Q5 Describe your process to integrate subcontractor design information and documentation into the system documentation? Identify the technical products and the set of software documentation you require as deliverables from your subcontractors. **C5**

Q6 What role does software documentation play in statusing the subcontractor's development activities? Describe how documentation is reflected in the SEMP/SEMS events and criteria. **C5**

Q7 How is this information reviewed and evaluated for adequacy? What are the criteria for complete documentation regarding both individual documents and the set as a whole? **C5**

Q8 Who within your organization is responsible to review and approve subcontractor software documentation? **C5**

Q9 What technical completion criteria for software are identified in the subcontract? Describe your test criteria and procedures for accepting subcontracted software? How is subcontracted software incorporated into your software integration and test process? **C6**

1.3.2 Subcontractor Development Management (cont.)

Q10 How is management visibility into the subcontracted development efforts established and maintained? Specifically, how do you assess a subcontractor's software development status? What is the basis for this assessment? How are Work Packages used in this assessment? What metrics/indicators are required from the subcontractor? How often is this information submitted and in what format? **C7**

Q11 What formal reporting do you require of your subcontractor relative to software? How is this reporting tied to the subcontractor WBS and in turn to your CWBS? To what level do you require the subcontractor's software to be identified and reported in his cost performance report? **C7**

Q12 How is software size control established and applied to subcontracted software development? **C8**

1 Program Management
1.3 Subcontractor Management
1.3.3 Subcontractor Planning

C1 The software development planning process includes subcontracted software development. **Q1**

C2 Program level Software Development Plan requirements and process requirements are levied on subcontractors. Subcontractor Software Development Plans are consistent with the prime's software development planning. **Q2**

C3 Subcontractor Software Development Plans are reviewed and approved by the prime contractor. **Q3**

C4 Procedures ensure that the program's development standards and procedures are applied to subcontractor development efforts or a process is in place to ensure that subcontractor standards and procedures are used which are compatible with the program's development processes. **Q3 Q4**

C5 The program's software documentation requirements and documentation approach are levied on subcontractors developing software. **Q2**

C6 If award fees or incentives are established for subcontractor developed software, measurable award fee or incentive criteria are established. **Q5**

Q1 How does your software development plan incorporate the planning for subcontracted software development? Describe specific planning coverage areas required to manage subcontracted software development. Is this planning based on a written organizational policy for managing software subcontractors? **C1**

Q2 Describe how the program's software development plan and process requirements including documentation requirements are flowed down to subcontractors developing software. How do you ensure that subcontractor software development plans are consistent with your software development planning? **C2 C5**

Q3 Are the subcontractor software development plans reviewed and approved? How are these plans incorporated into your subcontractor development monitoring and tracking activity? **C3 C4**

Q4 Do you apply your software standards to your subcontractor(s)? If not, what standards are required? How do you assure standards are followed and are compatible with the program's development processes? **C4**

Q5 Describe your approach to establishing award fees and incentives for subcontractor developed software. Are pre-defined criteria established? Describe the nature of these criteria. Do you plan the use of award fees or incentives on this contract? **C6**

1 Program Management
1.3 Subcontractor Management
1.3.4 Subcontractor Configuration Management

Criteria	Questions
C1 The subcontractors' software configuration management system is documented and is compatible and consistent with and supports the overall program software configuration management needs. **Q1**	**Q1** Describe how your subcontractor software configuration management system integrates with the overall program software configuration management system. Describe how they are mutually compatible and consistent. **C1**
C2 The subcontractors' software configuration management system is reviewed and verified by the prime contractor to be totally compliant with program requirements and needs, early in the development phase, i.e., prior to PDR. **Q2**	**Q2** How do you review your subcontractors' internal software configuration management system compliance to the program-level configuration management requirements? How early in the development is this compliance verified? **C2**

1 Program Management
1.4 Legal and Contracting Issues
1.4.1 Software Rights

C1 The software rights required for follow-on support and system maintenance including development System/Software Engineering Environments (S/SEEs) and tools are identified, including subcontracted and vendor software. **Q1 Q2**

C2 The identified restrictions to the use of the delivered software are compatible with the intended system operation and support concept. **Q1 Q2**

C3 The software development process is compatible with the limitations imposed by restricted rights and licensing restrictions for software. **Q2**

C4 The identified rights to reused software, either previously developed or concurrently developed across development teams, including subcontracted and vendor software on this program, are consistent with the program's life cycle support and maintenance needs. **Q1**

C5 The contractual terms proposed for use in acquiring subcontracted and vendor software includes provisions for the necessary rights to software and are compatible with the prime contractor's contractual obligations to the acquisition organization. **Q3 Q4**

C6 All delivered subcontracted and vendor software that is proposed as proprietary or restricted is fully justified in accordance with governing acquisition regulations (FAR, DFAR). **Q5**

Q1 Identify any proprietary or restricted rights software including reused software intended to be incorporated into the delivered system or in tools to be used to develop or support the delivered system. Provide in tabular form, the software identification, developer or vendor, and description of the associated restrictions. **C1 C2 C4**

Q2 Identify any restricted rights that would impact the ability to organically support the system and software or to competitively procure system and software support over the life cycle. Describe how your software development process accommodates these restrictions. **C1 C2 C3**

Q3 What contractual provisions have you instituted with your subcontractors to obtain rights sufficient for program life cycle support and maintenance? Describe how these contractual provisions meet your contractual obligations with the acquisition organization. **C5**

Q4 Where licensing agreements are required (e.g., on software tools), are they being negotiated for the total system program life cycle? **C5**

Q5 Provide for each proprietary or restricted rights software component a complete justification as to why this software is proprietary or restricted. **C6**

1 Program Management
1.5 Risk Control
1.5.1 Risk Identification

Criteria	Questions
C1 Short-falls and risks associated with the proposed development activities are identified. **Q1 Q2 Q3**	**Q1** Describe your process to identify and reduce technical risks associated with the system and software development. **C1**
C2 Critical paths and tasks in the software development and associated schedules are identified and monitored. **Q 4 Q5**	**Q2** Identify the projected risks and short falls associated with this program as a result of applying this process. **C1**
	Q3 What specific risks do you see in developing the subject program software with the selected design methodologies and implementation languages? **C1**
	Q4 Identify the critical tasks and paths associated with the proposed development plan. Describe your process to monitor these critical elements. **C2**
	Q5 Explain how you apply critical path and risk management techniques in managing software schedules. **C2**

1. Program Management
1.5 Risk Control
1.5.2 Risk Management

C1 Risk management strategies required to identify and reduce risk are identified consistent with the program's cost, schedule and performance baselines. **Q1**

C2 Identified risks areas are tracked and managed throughout the program development. **Q2**

C3 Specific criteria are identified in the risk management planning applicable to each risk reduction activity. These criteria define for each activity the condition under which each risk reduction activity is exercised. **Q3**

C4 Metrics exist for management tracking of specific program risk reduction actions. **Q4**

Q1 Describe your risk management process. What role will prototypes and demonstrations play in risk management? **C1**

Q2 Describe how identified risk areas are analyzed, tracked, and monitored throughout the program development. **C2**

Q3 How does your process support determining when to exercise the appropriate risk reduction activities? Are specific criteria established for each risk reduction activity? Are variance thresholds established for each risk area? **C3**

Q4 Identify and describe the metrics to be used to track specific program risk reduction activities. **C4**

2 Systems Engineering
2.1 System Requirements Development, Management and Control
2.1.1 Development and Allocation of Requirements

C1 A systems analysis and allocation process is used to verify that the performance and verification requirements are correct and complete at each level prior to further allocation and decomposition, and to verify them as to feasibility and top-level design concept prior to allocation to software. **Q1**

C2 The selected systems analysis and allocation methodology is compatible with other methodologies adopted on the program. **Q2**

C3 System requirements (including test and verification requirements) are analyzed, refined and decomposed to assure complete functional allocation to hardware and software. **Q3**

C4 When a system-level requirement is allocated to more than one configuration item (CI), a process is used to assure that the lower-level requirements taken together satisfy to the system-level requirement. **Q4**

C5 A defined process is used to generate the initial versions of the Software Requirements Specifications (SRS) and the Interface Requirements Specifications (IRS). A process to develop and review verification requirements for each performance requirement is in place. **Q5**

C6 A process exists to identify all design documents, requirements specifications, and interface specifications across the development team, including subcontractors. **Q6**

Q1 How are system and subsystem requirements defined and allocated? How are these requirements verified at each level prior to further allocation and decomposition? How are those requirements that imply digital processing and software verified as to feasibility and top-level design concept prior to allocation to software? **C1**

Q2 Describe how the systems analysis and allocation methodology is compatible with the systems design methodology, and with the software analysis methodology? **C2**

Q3 Describe the process by which system requirements are analyzed, refined and decomposed to develop a functional allocation to hardware, software, and other implementation technologies. Describe the process and specific trade studies and analyses performed to aid in deciding which requirements to allocate to hardware and which to software. **C3**

Q4 Describe the process which assures that when a system-level requirement is allocated to more than one configuration item (CI), the combination of the lower-level requirements meets the system-level requirement. **C4**

Q5 Describe the process that is used to generate the Software Requirements Specifications (SRS) and Interface Requirements Specifications (IRS). Describe the process to define verification requirements for each performance requirement as part of the requirements and definition (specification preparation) process. **C5**

Q6 Provide a tree diagram illustrating the hierarchical relationship among the various levels of system, subsystem, critical item, and prime item requirements specifications and design documents down to and including the software. This tree of design documents and specifications should include flowdown to subcontracted efforts. **C6**

2 Systems Engineering
2.1 System Requirements Development, Management and Control
2.1.2 Adequacy of Requirements

C1 System requirements are analyzed and refined to assure that they are consistent, clear, valid, feasible, compatible, complete, and testable, and they do not include inappropriate levels of design information. **Q1 Q2**

C2 Software Requirements Specifications (SRS) and Interface Requirements Specifications (IRS) are analyzed and refined to assure that all requirements allocated to software are adequately addressed, and that they do not include inappropriate levels of design information. They are reviewed by all affected parties. **Q1 Q3**

C3 If incremental development is planned, a process is used to establish functional, performance, and verification requirements for each incremental system or software block/build. This process assures all requirements are allocated to planned increments prior to the design and development of the increment. **Q4**

Q1 Describe the process used to assure that system requirements are consistent, clear, valid, feasible, compatible, and complete. How do you assure that inappropriate levels of design information are not contained in the requirements documents? **C1 C2**

Q2 Describe the process used to assure that system requirements have complete verification (test) coverage. **C1**

Q3 Describe the process by which the System Requirements Specifications (SRS) and Interface Requirements Specifications (IRS) are analyzed and refined to assure that all requirements allocated to software are adequately addressed. **C2**

Q4 If incremental development is planned, describe the process used to establish functional, performance, and test requirements for each incremental system or software block/build. Explain how these allocations to all planned blocks are reviewed and baselined prior to initiating the design and development of the first increment. **C3**

2 Systems Engineering
2.1 System Requirements Development, Management and Control
2.1.3 Requirements Change Control

C1 Requirements are baselined early in the program and are maintained under configuration control. **Q1**

C2 All changes to requirements, including those generated by the customer, are managed by means of a defined change process. **Q2**

C3 The allocation of new and additional requirements between hardware and software is managed by a structured change process; the reallocation of existing requirements between hardware and software is managed by a structured change process. **Q3**

Q1 Describe the requirements configuration management process. **C1**

Q2 Describe the requirements change control process, with reference to both internally and externally generated changes. **C2**

Q3 What process is used to control the allocation of changed (new or existing) requirements between hardware and software? **C3**

2 Systems Engineering
2.1 System Requirements Development, Management and Control
2.1.4 Software Impact Analysis

C1 The structured change process for requirements assures that the software impact for each proposed change is addressed. **Q1**

C2 All trade-off studies include an assessment of the software impact of each alternative; trade study results are documented and maintained for the life of the program. **Q2 Q3**

C3 Software is addressed in all systems engineering reviews. **Q4**

C4 Areas of the system with volatile requirements are monitored, and requirements specifications are reviewed for ambiguities that could result in software sizing and timing instability, and other program impacts. **Q5**

Q1 How is the software impact for proposed changes to system requirements addressed? **C1**

Q2 What process is used to include the software impacts of each alternative within system-level trade-off studies? **C2**

Q3 How are trade-off study results documented? How are they maintained? **C2**

Q4 What provisions exist to include software issues in systems engineering reviews? **C3**

Q5 How are areas of the system with volatile requirements monitored? Within those areas, how is the impact of potential requirements changes to the program (including software) identified and managed? **C4**

2 Systems Engineering
2.1 System Requirements Development, Management and Control
2.1.5 Requirements Traceability

C1 Two-way requirements traceability is maintained from system specifications to hardware and software configuration item specifications. **Q1**	**Q1** Describe the process used to provide two-way requirements traceability. At what point is requirements traceability established and documented? What provisions exist to maintain the traceability? **C1**

2 Systems Engineering
2.2 Computer System Architecture Design and Review Process
2.2.1 Architecture Definition

C1 A process exists for establishing and maintaining the computer system architecture (hardware and software), for determining the nature and number of the Computer Software Configuration Item (CSCI), and for maintaining traceability of the architecture to system requirements. **Q1**

C2 A process is used to define, maintain, and document interfaces (both internal and external) within the architecture. **Q2**

C3 A process is used to establish and show the relationships between the hardware and software components within the computer system architecture, including the system-level component hierarchy and control structure. **Q3**

C4 A process is used to establish and show the relationships between the computer system architecture (hardware and software) and the operational (human) interface. **Q4**

Q1 What process is used to establish and maintain the computer system architecture (hardware and software)? Describe how the nature and number of CSCIs is defined. Describe how the architecture is traced to system requirements. **C1**

Q2 Describe the process used to define, maintain, and document interfaces (both internal and external) within the architecture. **C2**

Q3 What process is used to establish and show the relationships between the hardware and software components within the computer system architecture, including the system-level component hierarchy and control structure? **C3**

Q4 What process is used to establish and show the relationships between the computer system architecture (hardware and software) and the operational (human) interface? **C4**

2 Systems Engineering
2.2 Computer System Architecture Design and Review Process
2.2.2 Adequacy of Architecture Design

C1 A process exists to evaluate how suitable the computer system architecture is for implementing all of the system functional and performance requirements, as well as how the design constraints are satisfied. **Q1 Q2 Q6**

C2 A process exists to evaluate the design based on the use of risk reduction techniques, such as the creation of models and prototypes (proofs, benchmarks). **Q4**

C3 A process exisits to periodically reassess the adequacy of the computer system architecture over the development cycle. **Q3 Q5**

Q1 Describe who participates in evaluating the adequacy of the computer system architecture design. **C1**

Q2 What process is used to evaluate how suitable the computer system architecture is for implementing all of the system functional and performance requirements? Within that process how are estimates made and budgeting done regarding the use of computer system resources? **C1**

Q3 What process is used to evaluate how suitable the computer system architecture is for meeting user needs? **C3**

Q4 Describe any plans for using risk reduction techniques such as the creation of models and prototypes (proofs, benchmarks). **C2**

Q5 What is the process for reassessing the adequacy of the architecture as the development of the system progresses? What criteria are used to either stay with the original design or change. **C3**

Q6 How is the performance of the architecture measured? How is the adequacy of the computational resources, memory, processor capacity, bus bandwidth established? **C1**

2 Systems Engineering
2.2 Computer System Architecture Design and Review Process
2.2.3 Architecture Design Review

Criteria		Questions	
C1	Whenever system requirements change there is a review of, and (as necessary) an update to, the computer system architecture design. **Q1**	**Q1**	Describe the review process for the computer system architecture design. **C1**
C2	The computer system architecture design is reviewed for flexibility to adapt to new system requirements. **Q2**	**Q2**	Explain the extent to which design-for-change considerations and flexibility to adapt to new system requirements are reviewed on the program, relative to the computer system architecture design. **C2**

2 Systems Engineering
2.2 Computer System Architecture Design and Review Process
2.2.4 Architecture Change Analysis

Criteria	Questions
C1 There is a review of all architectural changes and their impact on design margins (such as memory, throughput, bus loading and data latency) and cost and schedule baselines. **Q1 Q2**	**Q1** Describe the process used to review the impact of all architectural changes on design margins (such as memory, throughput, bus loading and data latency). **C1** **Q2** Describe the process used to review the impact of all architectural changes on cost and schedule baselines. **C1**

2 Systems Engineering
2.3 Supportability
2.3.1 Reliability

Criteria	Questions
C1 Reliability requirements are included in the system requirements, and are allocated to hardware and software. **Q1**	**Q1** Describe the process by which the system reliability requirements are allocated to hardware and software. **C1**
C2 Reliability is defined, measured, controlled and reported (in all life-cycle phases). A process is used to institute corrective actions when necessary. **Q2**	**Q2** How is reliability (in all life-cycle phases) defined, measured, controlled and reported? **C2**

2 Systems Engineering
2.3 Supportability
2.3.2 Maintainability

C1 Maintainability is defined, measured, controlled and reported. A process is used to institute corrective actions when necessary. **Q1 Q2 Q4**

C2 The support systems needed for any required operational self-sufficiency are developed, with an understanding between the customer and the developer regarding the effort, cost, and equipment required to support the system. **Q3**

Q1 How is maintainability defined, measured, controlled and reported? **C1**

Q2 How is maintainability built into the design? **C1**

Q3 Describe the process used to evaluate the effort, cost, and equipment needed to support the system. **C2**

Q4 Describe the process to manage corrective actions. **C1**

2 Systems Engineering
2.4 Intergroup Coordination
2.4.1 Group Interfaces

C1 Throughout the development life-cycle there is periodic coordination among developers, acquisition organizations, users, maintainers and testers regarding user needs, acquisition organization resources, technology status, and system requirements. Requirements changes that result from interactions with users, maintainers, and testers are managed with acquisition organization approval. **Q1 Q2 Q3 Q4**

C2 There is a systems engineering process which (as appropriate) emphasizes an integrated product development approach, and which defines the systems engineering interfaces with the other engineering disciplines and development activities, as well as the interfaces between the system and subsystem developers. **Q5 Q6 Q7**

C3 A process exists to manage, provide an escalation path for, and resolve conflicts regarding intergroup issues, including system-level issues that arise internally or with subcontractors. **Q8 Q9**

C4 Critical dependencies between development groups are identified and tracked. **Q10 Q11 Q12**

Q1 Describe the processes to be followed to have users and maintainers needs and viewpoints adequately reflected in system requirements throughout the development. **C1**

Q2 Describe the processes to be followed to keep system requirements in balance with acquisition organization resources throughout the development. **C1**

Q3 Describe the processes to be followed to have the system testers adequately involved in the requirements definition process throughout the development. **C1**

Q4 Describe the processes to be followed to assure that all requirements changes take place with customer approval. **C1**

Q5 To what extent is an integrated product development approach to be followed? **C2**

Q6 How will systems engineering interface with the other engineering disciplines and development activities? **C2**

Q7 How will interfaces between the various system and subsystem developers be managed? **C2**

Q8 Describe the processes for conflict resolution to be used internally between development groups. **C3**

Q9 Describe the processes for conflict resolution to be used between primes and subcontractors, and between subcontractors. Describe the processes used to identify and resolve intergroup product interface issues. **C3**

Q10 What critical dependencies exist between development groups? **C4**

Q11 Describe the processes for identifying new critical dependencies during the development effort. **C4**

Q12 How are critical dependencies between development groups tracked? **C4**

2 Systems Engineering
2.4 Intergroup Coordination
2.4.2 Tool Compatibility

C1 The support tools used by the different engineering groups enable effective communication and coordination. **Q1**	**Q1** Where different development groups have an interface, what support tools will be used to communicate and share data? Describe any areas of potential difficulty. **C1**

2 Systems Engineering
2.5 Systems Engineering Planning
2.5.1 Methodology and Standards

C1 Detailed systems engineering policies, practices and procedures are defined, consistent with systems engineering contractual standards. **Q1**

C2 The systems engineering process makes provisions for documenting the rationale of all major systems engineering decisions. **Q2**

C3 A process is used to arbitrate contention across trade-off studies for utilization of system-level resources and reserves. **Q3**

Q1 Describe how the program's systems engineering policies, practices and procedures are defined and documented. **C1**

Q2 Describe the provisions that have been made for documenting the rationale of all major systems engineering decisions. **C2**

Q3 What policies, practices and procedures govern system-level trade-off studies? When there is contention across trade-off studies for utilization of system-level resources, how are the competing claims resolved? **C3**

2 Systems Engineering
2.5 Systems Engineering Planning
2.5.2 Systems and Software Relationship

C1 Software engineering coordinates with systems engineering on all items that flow down to software engineering; for example, the system architecture, information models, and the identification, definition and allocation of software requirements. **Q1**

C2 A process is used to ensure that a staff with software skills conducts system-level trade-off studies for all issues that affect hardware and software and that they fully consider and account for software issues, including sizing, cost, schedule, memory, throughput, reuse, and other architectural considerations. **Q2**

Q1 Describe the role of software engineering on items that flow down from systems engineering to software engineering, such as the system architecture, information models, and the identification, definition, and allocation of software requirements. **C1**

Q2 How are software issues addressed in system-level trade-off studies? How does software engineering participate in those studies? How do you assure that your staff performing system level trade-off studies has adequate software skills? How do you assure that the system level trade-off studies account for software issues including sizing, cost, schedule, memory, throughput, reuse and other architectural considerations? **C2**

2 Systems Engineering
2.5 Systems Engineering Planning
2.5.3 SEMP/SEMS

C1 Systems engineering milestones (includingformal reviews) are defined and implemented with clear completion criteria in the SEMP/SEMS. **Q1**

C2 The Software Development Plan (SDP) is coordinated with and, as appropriate, incorporated into the SEMP/SEMS; all software milestones are accounted for. **Q2**

Q1 Describe the intended use of SEMP/SEDS/SEMS on the program. Are all major software milestones addressed in the Systems Engineering Management Plan (SEMP), the Systems Engineering Detailed Schedule (SEDS) and the Systems Engineering Master Schedule (SEMS)? Are completion criteria specified with all events? **C1**

Q2 Describe the relation of the Software Development Plan (SDP) to the SEMP/SEDS/SEMS. **C2**

2 Systems Engineering
2.5 Systems Engineering Planning
2.5.4 Staffing

C1 Within systems engineering there is a staffing plan that defines personnel requirements, including numbers, skill level, experience, and staffing profile. The systems engineering staffing plan identifies the personnel with required software skills and experience. **Q1**	**Q1** Is there a staffing plan for systems engineering on the program? Have required software skills and experience been identified? **C1**

2. Systems Engineering
2.5 Systems Engineeering Planning
2.5.5 Incremental Development

C1 Incremental software development, integration and test (with a series of builds), if used, is consistently integrated into any plans for system incremental development. **Q1**	**Q1** If there are plans for incremental software development, integration and test, describe how those plans are coordinated with any system incremental development plans. **C1**

2 Systems Engineering
2.6 System Integration and Test
2.6.1 Integration and Test Planning

C1 System integration planning begins in the early development stages of the program with the identification of the person responsible, and includes having that person participate in architecture and design reviews. **Q1**

C2 Test planning for each system build includes the multiple levels of system integration and test (from units to CSCIs to subsystem to system-level test). **Q2**

C3 Any incremental software development is incorporated into the system integration and test planning. **Q3**

C4 Any use of commercial-off-the-shelf (COTS) software or other reuse software is incorporated into system integration and test planning. **Q4**

Q1 When will the person responsible for system integration planning be identified? Does that person participate in architecture and design reviews? **C1**

Q2 If system builds are planned, describe how test planning for each system build includes the multiple levels of system integration and test (from units to CSCIs to subsystem to system-level test). **C2**

Q3 If incremental software development is planned, describe how it is incorporated into the system integration and test planning. **C3**

Q4 Describe any special integration and test plans developed for commercial-off-the-shelf (COTS) software or other reuse software. **C4**

2 Systems Engineering
2.6 System Integration and Test
2.6.2 Test Readiness

C1	Readiness criteria are clearly identified for formal subsystem and system test. **Q1**	**Q1**	Describe the process to be used to assure that readiness criteria are clearly identified for formal subsystem and system test. **C1**
C2	Readiness and completion criteria are clearly identified for informal subsystem and system test. **Q2**	**Q2**	Describe the process to be used to assure that readiness and completion criteria are clearly identified for informal subsystem and system test. **C2**
C3	The readiness criteria for both formal and informal test is incorporated into the SEMP/SEDS/SEMS milestones. **Q3**	**Q3**	How are these readiness criteria reflected in the SEMP/SEDS/SEMS? **C3**

2 Systems Engineering
2.7 Reuse
2.7.1 Opportunities to Reuse

C1 Opportunities to utilize previously developed system and software components (including architectures, designs, code, and documentation) are identified and subject to trade-off studies. **Q1 Q2**

C2 Opportunities to utilize non-developmental item (NDI), commercial-off-the-shelf (COTS), and government furnished equipment (GFE) system and software components (including architectures, designs, code, and documentation) are identified and subject to trade-off studies. **Q1 Q2**

C3 Opportunities for primes and subcontractors to utilize common system and software components in different parts (subsystems) of this development effort are identified and subject to trade-off studies. **Q3**

Q1 What system-level components that currently exist are candidates for reuse or adaptation on this development? What software components that currently exist are candidates for reuse or adaptation on this development? For each, indicate: source; whether NDI, COTS, or GFE; rights issues; availability; security issues; reliability level; likelihood of use; and projected percent of modification needed. **C1, C2**

Q2 What trade-off studies have been done or are planned to evaluate the costs, benefits, and risks of the opportunities to reuse existing system and software components? **C1, C2**

Q3 Describe any plans to identify and develop common system-level components for shared use across development groups or across configuration items (CIs). Describe any plans to identify and develop common software components for shared use across development groups or across configuration items (CIs). Include both intercompany and intracompany planning. **C3**

2 Systems Engineering
2.7 Reuse
2.7.2 Life-Cycle Issues

C1 The future reuse potential of newly developed system and software components is maximize within the constraints of the system cost, schedule, and performance baselines. **Q1 Q2**

C2 The life-cycle cost impact of reuse-related decisions is assessed, including: the choice of computer languages, processors, architectures, and environments; the development of reusable assets; and the maintenance of reuse repositories. **Q3**

Q1 Describe plans to make newly developed system and software components reusable (generic, adaptable). Include any efforts to perform domain engineering tasks. Describe the impact of these efforts on the system and software baselines. **C1**

Q2 Describe the impact of reuse plans on processes used to perform the systems and software engineering tasks (such as requirements analysis, design, implementation, integration and test). **C1**

Q3 Describe how the choice of computer languages, the choice of processors, the selection of system and software architectures, and the selection of operating and development environments relates to plans to create reusable system or software components, plans to reuse existing system or software components, or other reuse planning. **C2**

2 Systems Engineering
2.7 Reuse
2.7.3 Resource Management

C1 Processes, procedures and tools exist to document, manage and control reusable components. **Q1 Q2 Q3 Q4**	**Q1** Explain whether reusable components are documented differently from the rest of the system. **C1** **Q2** Explain whether reusable components are managed and controlled differently from the rest of the system. **C1** **Q3** Explain whether any special tools are needed to manage reusable components. **C1** **Q4** Explain whether any special processes or procedures are needed to manage reusable components. **C1**

3 Software Engineering
3.1 Software Development Planning
3.1.1 Software Estimating

C1 Estimates for the size, effort, cost, and schedule of each of the software components are generated according to a documented procedure. Estimates for incrementally developed software are generated consistent with published methods and company experience. **Q1**

C2 Estimates for the manpower profiles required for completion of the estimated schedules of each of the software products are generated according to a documented procedure. Estimated profiles are consistent with past actual company experience. **Q2**

C3 Estimates of the required critical computer resources needed by each of the software components are generated according to a documented procedure. **Q3**

C4 As software system definition progresses, estimates are performed for each of the lowest-level software components, and the previous estimates are revised. **Q4**

C5 Environmental parameters and calibration factors applied in estimating manpower requirements, and the resulting productivity factors, are consistent with past actual productivity rates for similar applications. **Q5**

C6 The estimating methods account for all tasks and steps associated with the software development process to be used. **Q2**

C7 All data required to repeat the above estimates for each of the software components are recorded and maintained. **Q6 Q7**

C8 The estimating process ensures consistency among estimates for size, workload effort, distribution of manpower, schedule and cost. **Q8**

C9 Software estimates are periodically compared to actual results to calibrate the estimating models and procedures. **Q9**

Q1 How are estimates for the size, effort, cost, and schedule of each of the software components generated? Which published estimating methods and models are used? Describe how estimates are developed for any planned incremental development or release? Describe your experience with this method relative to actual size, effort, cost and schedule of completed projects. **C1**

Q2 How are effort profiles estimated for the software components? What process ensures that each of the tasks required for the software development (i.e., requirements, definition, analysis, design, code, integration, test) are included in the estimates? How are the cost and effort of related software engineering tasks (such as configuration management, quality assurance, and test and integration) included in the estimates for the program?**C2 C6**

Q3 How are estimates generated for required critical computer resources needed by each of the software components? How are the computer resources estimated and balanced across the program to ensure critical needs are met? **C3**

Q4 Describe how the estimates are revised as software system definition progresses. What process ensures that the estimates are kept consistent with the current state of the overall program? What is the approach to revising estimates as lowest-level software components are comprehensively defined? By what program milestone are these revised estimates baselined? **C4**

Q5 How are the various environmental parameters and calibration factors derived? What is the source of the productivity factors applied in estimating manpower requirements? Are they consistent with past actual productivity rates? Are environmental parameters and calibration factors for use in estimating under configuration management? **C5**

3.1.1 Software Estimating (cont.)

Q6 How is the data required to repeat the above estimates for each of the software components recorded and maintained? Is the data configuration controlled and available to all who need it? Areoccasional audits done to verify that the required data is accurate and available? **C7**

Q7 Who has the responsibility for development and storage of the above estimates? Who ensures that estimates are done according to procedure, and that the data is recorded and maintained? **C7**

Q8 Describe the methodology for correlating and ensuring consistency among estimates of size, workload effort, distribution of manpower, schedule and cost. **C8**

Q9 How is the accuracy of the software estimates ensured? Is there a formal process for periodically calibrating the software estimating procedures with actual performance data? **C9**

3 Software Engineering

3.1 Software Development Planning

3.1.2 Software Work Packages

C1 Software components and work packages of manageable size and development effort are defined to enable management of the entire software system. **Q1**

C2 The defined set of software work packages is used to manage the work tasks associated with software development. **Q2**

C3 Manpower is allocated to the individual work packages consistent with their individual development schedules. In addition, software development manpower is allocated consistent with the total software development needs of the program. **Q3**

Q1. How is the overall software effort organized into manageable software components? What factors are considered in determining the appropriate size and development effort for each of the components? How is the software organization documented? **C1**

Q2 How are software work packages planned and defined? Describe the criteria for acceptable software work packages. Explain how the software work package is used to manage the work, i.e., plan, define, assign resources and responsibility, status and report progress. **C2**

Q3 Describe your method for allocating manpower to individual software work packages as well as across the total software development effort. How does this allocation method apply to incremental (block, build) software development. How does the method provide realistic manpower profiles, based on experience, to support the total program needs? **C3**

3 Software Engineering
3.1 Software Development Planning
3.1.3 Software Engineering Development Methods

Criteria		Questions	
C1	An engineering development life-cycle model consistent with the program requirements and needs is selected. The program's software development process is integrated into the selected system's engineering development process. **Q1**	**Q1**	Describe how a systems engineering development life-cycle process is selected consistent with the program requirements and needs. How is the program's software engineering development process integrated into the selected systems engineering development process? **C1**
C2	Software engineering development methodologies consistent with the programs' requirements and needs are selected and integrated into the programs' software development process. These methodologies are supported across the entire life cycle. **Q2**	**Q2**	Describe how new or different software development methodologies are selected and integrated into the program's software development process? How are such methodologies supported across the entire program life-cycle? **C2**
C3	The rationale for selecting the life cycle development models and methods is recorded and maintained. **Q3**	**Q3**	Where is the rationale for selecting the new or different methodologies recorded and maintained? How are the lessons learned during the insertion of a new or different methodology recorded and available for reference? Who has the responsibility for maintaining organizational records on the insertion of new development methodologies? **C3**

3 Software Engineering

3.1 Software Development Planning

3.1.4 Preparing the Software Development Plan

C1 The development of the SDP is compliant with the RFP and contractual requirements and is consistent with company standards for SDP preparation. **Q1**

C2 A comprehensive integrated plan for the entire program software development is written into the SDP. This plan covers all software to be developed and is consistent with the magnitude and complexity of the development effort. **Q2**

C3 The subcontractors' software development process is comprehensively defined in an SDP. **Q3**

C4 All of the involved parties within the software development organization participate in the generation of the SDP, and demonstrate understanding and commitment to its terms. The SDP is coordinated throughout the program organization, including subcontractors. **Q4**

C5 The defined software development process is designed to meet the specific needs of the program, either by tailoring the organization's standard software processes, or by merging the processes used by the prime and subcontractors into a cohesive, integrated development process. **Q5**

C6 The development process documented in the SDP identifies major events that are included in the SEMP/SEMS (e.g., CDR, PDR, SSR, etc.). **Q6**

C7 The SDP comprehensively describes processes, schedules, and manpower for any planned incremental development that is consistent with program requirements and needs. **Q7**

C8 For programs involving software developed by primes, associates, and subcontractors, common and consistent software processes are established. **Q8**

Q1 Describe the process used in developing the SDP. What company standards define the requirements for and process of preparing an SDP? What data are used as a basis for the development of the SDP? What is required to be covered by the SDP? How is it ensured that the SDP is consistent with the SEMP and other system- or software-level requirements? **C1**

Q2 Describe (outline) the coverage in the SDP. Explain how the SDP comprehensively describes a cohesive plan for the entire software development consistent with the complexity and magnitude of the software development effort. How is it ensured that all software is covered in the SDP? **C2**

Q3 How is the subcontracted software process development planned and comprehensively described in the SDP? How do subcontractors buy into and commit to their parts of the SDP? What contractual mechanisms exist to ensure subcontractor compliance with their SDP? **C3**

Q4 How do all of the components of the software development organization participate in generating the SDP? How do they demonstrate understanding and commitment to the terms of the SDP? Which organizations, including subcontractors, coordinate on the SDP? How are the terms, dependencies, and responsibilities negotiated and communicated, both internal to the prime and among the various subcontractors? **C4**

Q5 Explain the relationship between the program software development processes and the company standard software development processes. For new processes, please describe the rationale for selecting the new processes and any advantages or risks associated with the new processes. Identify previous experience with new or nonstandard processes to be applied on the program. Describe how the development processes are consistent at key interface points. **C5**

3.1.4 Preparing the Software Development Plan (cont.)

Q6 Identify the major software engineering development events and reviews described in the SDP. How do these events and reviews interface and support the systems engineering events and reviews in the SEMP/SEMS? Are specific SEMP/SEMS entrance and exit criteria defined in the SDP? How does the program SDP reflect the engineering process requirements? **C6**

Q7 How does the SDP accommodate and specifically describe the planning, processes, schedules and manpower to support any planned incremental software development (blocks/builds)? **C7**

Q8 How are common software development processes determined among primes, associates, and subcontractors to ensure continuity, integrity and life cycle supportability of software? **C8**

3 Software Engineering
3.2 Software Project Tracking and Reporting
3.2.1 Software Tracking

C1 The size, effort, cost and schedule status of each of the software work packages is periodically measured and reviewed by engineering management and corrective actions are taken when pre-established variance thresholds are exceeded. **C1**

C2 The critical computer resources required by and allocated to each of the software work packages are periodically measured and reviewed by management, and corrective actions are taken when pre-established variance thresholds are exceeded. **C2**

Q1 How often will engineering and program management measure and review the size, effort, cost and schedule status of each of the software components? What criteria and conditions will trigger corrective actions? How will the success of the corrective actions be measured? What provisions exist for event-driven engineering management reviews? **C1**

Q2 How often will the critical computer resources required by each of the software components be measured and reviewed by engineering and program management? When will it be deemed necessary to take corrective actions? Who has responsibility for setting the variance thresholds? **C2**

3 Software Engineering
3.2 Software Project Tracking and Reporting
3.2.2 Software Reporting

C1 The status of each software work package is reported to all involved levels of engineering and program management through periodic reporting up the chain of command. **Q1**

C2 Development process/performance and product quality measurements are recorded, analyzed, and used for improving process and product quality on the program. These data are recorded and maintained for organizational process and product quality improvements. **Q2**

Q1 How is the status of each of the software work packages reported up the chain of command? What specific elements of software status, e.g., units, components, configuration items, subsystem, system, are reported to each management level from first-level supervisor through the program manager. What situation, condition, threshold, would trigger a status report to a higher level of management than would normally be necessary for a work package? **C1**

Q2 What actual measurements of development performance and product quality will be recorded during software development? How will these measurements be analyzed and used for changing and improving the products and processes? How will the metrics be recorded and maintained? Who is responsible for the collection, storage, and analysis of metrics? **C2**

3 Software Engineering
3.3 Software Requirements Management
3.3.1 Software Requirements Analysis

C1 The software requirements are analyzed for completeness, correctness, clarity, feasibility and verifiability. **Q2 Q3**

C2 Requirements that are derived from the Software Requirements Specification are documented and maintained. **Q4**

C3 The selected requirements analysis methodology is compatible with other methodologies applied on the program. The analysis methodology is supported with necessary tools. **Q1 Q5**

Q1 Describe the software analysis process to be applied. Identify the specific methodologies and tools to be used to support the analysis process. What organizational element is responsible to perform the analysis? Identify the input to and output product from the analysis. **C3**

Q2 What are the software requirements analyzed for, i.e., completeness, correctness, etc. How do you determine that the software requirements are complete, adequate, and verifiable? **C1**

Q3 How are the total sets of software requirements analyzed as a whole, including interfaces? **C1**

Q4 If additional requirements are derived from the baselined requirements, where are they documented? How are they maintained? How is there impact on cost and schedule determined? **C2**

Q5 Describe the methodology used to analyze the requirements. Is it compatible with the requirements traceability methodology? Is it compatible with the design methodology? Is it compatible with the development language? **C3**

3 Software Engineering
3.3 Software Requirements Management
3.3.2 Software Requirements Changes

C1 The software development artifacts (requirements, design, code, documentation) are revised as changes to the requirements are incorporated. **Q1**	**Q1** Describe the software development activities that result from a change in or addition to the requirements. When do they get performed? How do you ensure that they are performed? **C1**
C2 As changes and additions to the requirements are incorporated the software development plans and program baselines (cost, schedule) are reviewed and modified if necessary. **Q2**	**Q2** Describe the software planning activities that result from a change in the requirements. When do they get performed? How do you insure that they are performed? **C2**
C3 Two-way traceability between the software requirements and the system requirements is established and maintained. **Q3**	**Q3** How do you ensure that two-way traceability between the system requirements and the software requirements is maintained? **C3**

3 Software Engineering
3.4 Software Design
3.4.1 Design Methodology

C1 A methodology is used to develop, document and maintain the top-level and detailed software design. **Q1**

C2 The design description includes the (static) structure and the (dynamic) behavior of the software. **Q2 Q3 Q4**

C3 The design description includes all the software interfaces. **Q5**

C4 The estimated use of the target computer resources is refined and included in the design documentation. **Q6 Q7**

C5 Plans and mechanisms exist to document the design decisions including tradeoff studies. **Q8**

C6 The selected design methodology is compatible with other methodologies adopted on the program. **Q9 Q10**

C7 The taxonomy used to represent design, code, and test entities are either common across the entities or are compatible and are mapped. **Q7 Q10**

Q1 Describe the process and specific methodologies used to develop the top-level and detailed software design. Is the same methodology used to maintain the design through development and life cycle support? What tools are used to support the methodology? **C1**

Q2 What mechanism and format are used to describe the components of the design? **C2**

Q3 What mechanism and format are used to describe the execution priorities of the different components, and the execution control? **C2**

Q4 How is the data flow described? **C2**

Q5 How does the design methodology describe the interfaces internal to the software components? How does it describe the interfaces between the software components? How does it describe the interfaces between the software and other components of the system? **C3**

Q6 Describe the timelines generated to represent the CPU usage of the target computer(s)? How are these timelines maintained with the design documentation? **C4**

Q7 Describe the memory maps (or other representations of memory utilization) generated to represent the memory usage of the target computer(s)? How are they maintained with the design documentation? **C4 C7**

Q8 How are the design decisions documented and communicated? How are the trade-off results analyzed? How is this process enforced? **C5**

Q9 Is the design methodology compatible with the requirements analysis methodology? Is it compatible with the development language? **C6**

Q10 Is the same taxonomy (CSCs, Units, Packages, Files, etc.) used to represent the design entities, the code entities, the test entities and the configuration management entities? If not, how is the mapping between them defined? **C6 C7**

3 Software Engineering
3.4 Software Design
3.4.2 Design Assurance

C1 Mechanisms exist to ensure the modularity, cohesiveness, feasibility, and coupling of the design. **Q1 Q2 Q3**

C2 Exit criteria are used to ensure the completeness and quality of the design before it is baselined. **Q4**

C3 Two way traceability between the design and the requirements is established and maintained. **Q5**

Q1 Describe the exit criteria for baselining the top-level design. Do they include adherence to the design methodology? What is the criteria for completeness of the design documentation? How are the development standards required for the program reflected in the exit criteria? How are they enforced? **C1**

Q2 Describe the exit criteria for baselining the detailed design. Do they include adherence to the design methodology? What is the criteria for completeness of the detail design documentation? How are the development standards required on the program reflected in the exit criteria? How are they enforced? **C1**

Q3 How is the correctness, completeness, and feasibility of the design ensured? Who has that responsibility? What tools are used? Is it part of the exit criteria described above? Identify the design quality attributes that are included in the exit criteria. **C1**

Q4 How are the modularity, Cohesiveness and coupling of the design ensured? Who has that responsibility? What tools are used? Is it part of the exit criteria described above? Identify the design characteristics that are included in the exit criteria. **C2**

Q5 How is traceability established from the requirements to the design, and from the design to the requirements? At what point in the design is it done, and by whom? How is it documented? How is it maintained? What tools are used? Is this traceability part of the exit criteria described above? **C3**

3 Software Engineering
3.5 Software Coding and Unit Testing
3.5.1 Code Development

C1 The program software coding standards contain guidelines regarding internal documentation, style, complexity and use of language features. **Q1 Q2 Q3 Q4**

C2 The use of the target computer's resources is measured and compared to budgeted values. Corrective action is taken as required. **Q5 Q6**

C3 The developed software is unit tested. Realistic resources and schedules are allocated to this level of testing. Units are tested in all increments of development. **Q7 Q8**

C4 The software is reviewed against the design, and two-way traceability between the software code and the design is established and maintained. **Q9**

C5 Exit criteria exist for establishing that each lowest-level software unit has been implemented correctly, is performance tested and is in conformance with the coding standards. **Q10**

Q1 Describe the process for ensuring that the coders use a common coding style and that the code is sufficiently and uniformly documented. **C1**

Q2 Explain how the complexity of the code is kept to a minimum. What guidelines are followed? What tools are used? **C1**

Q3 Identify any guidelines for the use of special features of the development language? How are they communicated? **C1**

Q4 Describe any plans to use code generation technology for this development. What are the potential benefits, risks, and life-cycle trade-offs? **C1**

Q5 How are timing measurements made? At what component level? How often? Who has that responsibility? How are the results used? **C2**

Q6 How are memory usage measurements taken? At what component level? How often? Who has that responsibility? How are the results used ? **C2**

Q7 What processes and procedures are used to ensure that the design is implemented completely and correctly? At what component level? Who has that responsibility? **C3**

Q8 How do you ensure that each lowest-level software component (unit) is unit tested? How do you ensure that adequate resources are allocated and adequate schedule (duration) is planned to support this level of testing? What tools are used? Who has that responsibility? **C3**

Q9 How is the traceability from the software code to the design and from the design to the code established and maintained? When is it done? How is it documented and maintained? What tools are used? Who has that responsibility? **C4**

Q10 What exit criteria exist for establishing that each lowest-level software unit is ready for integration? Do they include compliance with the coding standards? Do they include, peer reviews? Do they include unit testing? Do they include conformance to the design? How are they enforced? **C5**

3 Software Engineering
3.5 Software Coding and Unit Testing
3.5.2 Code Changes

Criteria		Questions	
C1	Code changes are unit tested before they are incorporated. **Q1**	**Q1**	Describe your process for ensuring that all code changes are unit tested. **C1**
C2	Code changes are reviewed for correctness, and to avoid undesired impact on other software and system variables and components. **Q2 Q3**	**Q2**	Describe your process for estimating the effect of code changes on other parts of the system, including variables and other software components. What tools are used? Who is involved in the process? **C2**
		Q3	Describe your process for reviewing code changes after the code has been baselined. Are they reviewed for correctness, and compliance with the standards? What if they are urgent changes? How is the review process enforced? Who has that responsibility? **C2**

3 Software Engineering
3.6 Software Integration and Test
3.6.1 Software Integration

C1 The software integration planning takes into account the interdependencies between the different software components and the criticality of each component . **Q1 Q2 Q3**

C2 The software integration planning takes into account the availability of other components of the system. **Q1 Q4**

C3 For planned incremental software development, the software integration is planned, scheduled, and resources are allocated to support each increment of software development. **Q5**

C4 The software integration planning and process accommodate software integration starting with the lowest level elements, i.e., units through all levels, including CSCI and CSCI/HWCI. **Q1**

Q1 Describe your process for planning the software integration. How many different components do you integrate at once? How do you determine the order for integrating the different software components? Describe how your integration process accommodates all levels of software integration. **C1, C2, C4**

Q2 How are the dependencies between the different software components determined? At what level? How does it affect integration planning? **C1**

Q3 How is the criticality of each component determined? What role does it play in integration planning? **C1**

Q4 How does your integration planning handle situations where a needed software or hardware component is not available on time? **C2**

Q5 How does your integration planning cover integration with all planned software increments (blocks, builds). **C3**

3 Software Engineering
3.6 Software Integration and Test
3.6.2 Software Testing

C1 The software test process includes development of test plans, procedures, and test cases. **Q1**

C2 A process exists to ensure that software testing is adequately planned with sufficient test resources. **Q2**

C3 An approach is used that plans for all levels of testing to ensure thorough testing of the software. **Q3 Q4**

C4 A process exists for incorporating changes resulting from software testing. **Q5 Q6**

C5 A regression test methodology ensures that system performance is maintained after revisions are made to the software components. **Q7 Q8**

C6 Software testing is planned with adequate schedules and resources for all planned software development increments (blocks, builds). **Q9**

Q1 How are test plans, test procedures and test cases developed? When ? By whom? Where are they documented? How are they reviewed? How are they controlled? **C1**

Q2 What tools will be used for testing? When will they be available? Will they require any special inputs? Will their outputs require any special processing? What is your process to ensure that all required test resources have been planned and allocated? **C2**

Q3 Does your software test and verification process define specific levels of software test ? What are they? How do they relate to the structure of your software design? **C3**

Q4 What are the completion criteria for each level of testing? Do you generate test plans, and test procedures for each level? If so, how are they coordinated across the different levels? **C3**

Q5 Describe your process for incorporating changes resulting from errors that are uncovered during testing? Where is it documented? How is it enforced? **C4**

Q6 Does your change process ensure that the changes get incorporated into the right baseline version ? Does it have provisions for priority changes that require quick turnaround? **C4**

Q7 What is your process for regression testing? Are there guidelines for when and how the regression tests should be run? Is regression testing factored into the schedules? **C5**

Q8 Is there a library of regression tests? If so, describe how it is generated. Are any regression testing tools used? **C5**

Q9 How is the software test process and discipline applied to each planned increment (block, build) of software developed? **C6**

4 Quality Management and Product Control
4.1 Software Quality Management
4.1.1 Quality Planning

C1 The program's software quality plan is the basis for the program's activities for software quality management. **Q1 Q2 Q4**

C2 The program's software quality plan contains provisions to ensure that quality is built into the software. **Q3**

C3 The program's quality goals and their priorities for the software products are defined, monitored, and revised throughout the software life cycle. **Q5**

C4 The plan identifies points in the process where software quality is measured. **Q6**

C5 The plan identifies methods for analyzing the program's quality measurements, for evaluating whether they meet the customer's needs, and for determining the necessary corrective actions. **Q7**

C6 The software program's quality goals for the products are allocated appropriately to the subcontractors delivering software products to the program. **Q8**

C7 Plans exist for the members of the development organizations to receive required training in software quality. **Q4**

Q1 Describe your process for developing a quality plan for the program. **C1**

Q2 How are the activities described in the program quality plan reflected in proposed work packages? **C1**

Q3 What provisions does the quality plan have to ensure that quality is built into the software? **C2**

Q4 Is there quality training planned for the development organizations? Is it mandatory? What does it consist of? **C1 C7**

Q5 Does the quality plan define specific quality goals for each software product? Does it describe how these goals are prioritized? Does it describe how these goals are monitored and kept consistent with the customer's needs? **C3**

Q6 Are there checkpoints for measuring software quality identified in the quality plan? **C4**

Q7 Does the quality plan describe how the quality data is analyzed, and how it is used? **C5**

Q8 Have the program's quality goals been incorporated into subcontracts where appropriate? **C6**

4 Quality Management and Product Control
4.1 Software Quality Management
4.1.2 Product Evaluations

C1 Independent product evaluations are performed for all software work products before they are baselined. **Q1 Q2 Q3 Q4 Q5**

C2 Responsibility for each product evaluation is clearly defined. **Q5**

Q1 Do you have evaluation procedures for all the different software products that will be developed on the program ? Do they describe the evaluation criteria in sufficient detail? **C1**

Q2 When are the different software products evaluated? **C1**

Q3 Are adequate resources (cost and schedule) provided for them ? **C1**

Q4 What are the qualifications of the evaluators? Do they receive any special kind of training? **C1**

Q5 Is the responsibility for each product evaluation defined? At what level? **C1 C2**

4	Quality Management and Product Control
4.1	Software Quality Management
4.1.3	Software Discrepancies

C1	Specific procedures exist to resolve software versus hardware discrepancies and to identify, document, track, and resolve software discrepancies. **Q1 Q2 Q3 Q4**	**Q1**	How are the software discrepancies managed? **C1**
		Q2	Identify and describe specific procedures to identify, document, report, track, and resolve software discrepancies. **C1**
		Q3	Describe your method for resolving software versus hardware discrepancies in your problem reporting systems. **C1**
		Q4	Describe tools used, and any automation, to address discrepancy collections, tracking and reporting. **C1**

4 Quality Management and Product Control
4.2 Software Quality Assurance
4.2.1 SQA Organizational Approach

C1 An organization is assigned the responsibility to monitor the software development process, and the software products. **Q1**

C2 The responsibilities, mission and interface(s) of quality assurance with the engineering, configuration management, and test functions are defined and documented. **Q1 Q2 Q3**

C3 The SQA group is empowered to effect changes to the program when quality goals are not adhered to. **Q4 Q5**

Q1 Describe the responsibilities of the SQA organization and how SQA interfaces with other organizations. **C1 C2**

Q2 Does the SQA organization communicate the results of SQA activities to the engineering organization? **C2**

Q3 How does the SQA function interface with engineering, configuration management and test functions? **C2**

Q4 What can SQA organization do if the software development process and procedures are not being followed? **C3**

Q5 What mechanisms and channels exist for SQA to surface quality problems and elevate them in the management chain until they are resolved? **C3**

4 Quality Management and Product Control
4.2 Software Quality Assurance
4.2.2 SQA Staffing

C1 Sufficient QA personnel are staffed to the program to accomplish their assigned responsibilities and functions as proposed for this program. **Q1**

C2 Qualified QA personnel are assigned. **Q2**

Q1 How many software QA people are normally assigned to a major program? What percentage of the software budget is expended on software QA activities? **C1**

Q2 What are the required qualifications of SQA personnel? Do they receive training in software development processes? Do they receive any program-specific training? **C2**

4 Quality Management and Product Control
4.2 Software Quality Assurance
4.2.3 Compliance Checking

C1 The program follows a written SQA plan for measuring and monitoring the performance of the program's defined software process. **Q1**

C2 Adherence to the defined software development and management processes is verified. **Q2**

C3 SQA audits designated software work products to verify compliance with quality goals, and adherence to the applicable standards, and requirements. **Q3–Q4**

Q1 Where are the SQA activities defined for the program? **C1**

Q2 Describe how SQA ensures compliance of the software development activities with the defined processes. Which processes are audited? How often? **C2**

Q3 Describe how SQA ensures compliance of the software management activities with the planned processes. Which processes are audited? How often? **C3**

Q4 Describe how SQA verifies that the software products adhere to the program's requirements, standards, and quality goals. **C3**

4 Quality Management and Product Control
4.3 Defect Control
4.3.1 Defect Activity Coordination

C1 The program develops and maintains a plan for its defect prevention activities. **Q1**

C2 Revisions to the standard software process resulting from defect prevention actions are incorporated. **Q2**

Q1 Describe your program plan for preventing software defects? **C1**

Q2 Are defect causes assessed for potential process improvement and incorporation into program and organizational development processes? **C2**

4 Quality Management and Product Control
4.3 Defect Control
4.3.2 Defect Collection and Analysis

C1 Common causes of defects are identified, prioritized and systematically eliminated. **Q1 Q2 Q3** **C2** Causal analysis meetings are conducted. **Q1 Q4** **C3** Data on defects identified in peer reviews, document review and testing are collected and analyzed. **Q5**	**Q1** Describe your approach to the collection and analysis of defects. **C1 C2** **Q2** Does your program have a process for identifying common causes of defects? **C1** **Q3** Are known common causes of defects prioritized for correction? **C1** **Q4** Are casual analysis meetings conducted? Define your procedures for conducting causal analysis meetings? **C2** **Q5** Identify your approach to collecting defects resulting from peer reviews, testing, and design reviews. Is this approach contained in the quality plan? **C3**

4 Quality Management and Product Control
4.3 Defect Control
4.3.3 Defect Reporting

C1 Defect prevention data are documented and tracked across the teams coordinating defect prevention activities. **Q1** **C2** Members of the software engineering group and software-related groups receive feedback on the status and results of the organization's and program's defect prevention activities on a periodic basis. **Q1**	**Q1** Is defect prevention data documented and communicated across all teams participating in defect prevention activities? Does this information include feedback on the status and results of defect prevention activity? What is the frequency of this communication? **C1 C2**

4 Quality Management and Product Control
4.4 Metrics
4.4.1 Metrics Definition and Collection Process

C1 The metrics selected and the strategy for the data collection and the analyses to be performed are determined based on the program's defined software process. **Q1 Q2 Q3**

C2 The specific measurement data to be collected, their precise definitions, the intended use and analysis of each measurement, and the process control points at which they will be collected, reported and fed back are defined. **Q2 Q3**

C3 The established metrics process includes the requirement to define variance thresholds, which when broken require corrective action. **Q4**

Q1 Describe your process for defining the set of metrics that will be used on this program and determining the sue of these metrics. **C1**

Q2 Describe for each collected metric, how it will be collected, the points at which it will be collected, how it will be analyzed, how it will be reported, and to which organizations it will be reported. **C1 C2**

Q3 Describe the processes and tools used for data collection and analyses. **C1 C2**

Q4 Describe your use of variance thresholds. Describe how these thresholds are established and used in the management of the development. **C3**

4 Quality Management and Product Control
4.4 Metrics
4.4.2 Metrics Selected for the Program

C1 The metrics selected for the program address the system and software products, the process used to generate the products, and the progress of the development effort. **Q1**

C2 The measurement program is integrated with the program's development process, across the life-cycle and the teaming arrangements. **Q2 Q3 Q4**

Q1 Identify the metrics you plan to collect on this program, which system or software product they apply to, which process they apply to and or what progress they measure. **C1**

Q2 Describe for each collected metric what life-cycle phase it applies to. **C2**

Q3 Describe the different organizations that will be involved in the measurement program, what will their role be and how they will interface. **C2**

Q4 How are these measurements defined and integrated with your program's defined software process? **C2**

4 Quality Management and Product Control
4.5 Peer Reviews
4.5.1 Peer Review Planning

C1 Internal documents exist that: identify required participants in the reviews, provide specific criteria for successful completion, describe documentation required for the review and describe how follow-on actions are documented, tracked and controlled. **Q1**

C2 Peer reviews are planned consistent with the peer review internal standards and procedures. **Q2**

C3 Peer Review Plans specify the schedule of peer reviews. **Q2**

C4 The Peer Review Schedule is consistent with the SEMP/SEMS. **Q2**

Q1 Describe the documented internal peer review procedures and requirements including definition of required participants, completion criteria and review content and follow-on action item resolution. **C1**

Q2 Describe how peer reviews are planned and scheduled. Describe how the peer review schedule is consistent with other program schedules (e.g. SEMP/SEMS). **C2 C3 C4**

4 Quality Management and Product Control
4.5 Peer Reviews
4.5.2 Peer Review Performance

C1	Peer reviews are performed according to the peer review plan. **Q1**	**Q1**	Describe how peer reviews are performed according to the peer review plan. **C1**
C2	The reviews are documented, i.e., the review process, requirements, conduct and results. **Q2**	**Q2**	Describe how peer review results are documented and to whom the results are distributed. **C2 C3**
C3	Review results are reported to the appropriate managers. **Q2**		

4 Quality Management and Product Control
4.6 Internal Independent Verification and Validation (IIV&V)
4.6.1 IIV&V Planning

C1 A well defined systematic approach to internal independent software verification and validation is documented in accordance with contract requirement. **Q1**

C2 The planning process for IIV&V includes criteria to select elements of the software for which the IIV&V process is applicable. **Q2**

C3 Elements of the software to which to apply the IIV&V process are selected. **Q2**

C4 Required resources and tools are identified in the IIV&V planning. **Q2**

Q1 Describe your process for planning and conducting IIV&V? **C1**

Q2 Does the organization have a procedure for determining which software should undergo IIV&V? Has this procedure been applied to identification of software for IIV&V for this program? **C2 C3 C4**

4 Quality Management and Product Control
4.6 Internal Independent Verification and Validation (IIV&V)
4.6.2 Technical Evaluation and Implementation Process

C1 The IIV&V function is independent of the software development function. **Q1**

C2 Sufficient resources and required tools are available to accomplish the IIV&V process. **Q2**

Q1 What is the organizational responsibility and reporting chain for the program IIV&V? **C1**

Q2 Describe the resources and tools available to accomplish the IIV&V process. **C2**

4 Quality Management and Product Control
4.7 Software Configuration Management
4.7.1 SCM Planning

C1 SCM is organized as an integral part of the system CM process and is integrated with engineering, program management and other development disciplines. **Q1 Q2**

C2 A process exists for the development, maintenance, and distribution of the program's SCM plan, standards, and procedures. **Q3**

C3 An approved SCM plan is used as the basis for performing the SCM activities. **Q4**

C4 The SCM Planning requires creation and management of the program's software baseline library. The baseline library contains the functional, allocated, developmental and product baselines. **Q5 Q6**

Q1 Describe how the software configuration management is integrated with the system CM process? **C1**

Q2 Describe how software configuration management is integrated with engineering, program management and other development disciplines? **C1**

Q3 What guidance exists for the development, maintenance, and distribution of the program's SCM plan, standards, and procedures? **C2**

Q4 Is there a software configuration plan for this program? Who reviews and approves the plan? **C3**

Q5 Does the CM plan address control of the functional, allocated & developmental and product baselines? **C4**

Q6 Does the CM plan require creation and management of a program software baseline library? Where are the library procedures documented? **C4**

4 Quality Management and Product Control
4.7 Software Configuration Management
4.7.2 Baseline/Configuration Identification and Management

C1 The configuration control implementation establishes a developmental configuration for each CSCI, controls the preparation and dissemination for changes to the master copies of deliverable software and documentation and maintains current copies of deliverable documentation and code. **Q1**

C2 The CM process includes establishing, documenting and controlling the functional, allocated, developmental and product baselines and how these baselines are configured as a system with the hardware baselines. **Q2 Q3**

C3 Products from the software baseline library are created and released according to a procedure. Procedures exist for management of:
- software requirements document(s)
- software design document
- code
- test plans, test procedures, and test cases
- test results

Q4

C4 Procedures exist and are followed to create and maintain developmental builds and incremental test baselines. **Q2 Q5**

C5 The software work products to be placed under configuration management are identified. **Q6**

C6 The software hierarchical structure uniquely identifies elements of the system consistently across all disciplines. **Q7**

C7 Software specification versions, and revisions for each release, are formally identified within the SCM system. **Q8**

C8 Each element of the software system is uniquely identified by name and number.
Q9

Q1 How are software baselines, both formal and informal, controlled using documented procedures for software and documentation and for transfer to other libraries, where appropriate? **C1**

Q2 How are the software functional, allocated, development and product baselines established, documented, and controlled? Explain how baselines are established and controlled for each block or build (incremental software development). **C2 C4**

Q3 How are hardware and software baselines configured as a system? **C2**

Q4 Explain the SCM library procedures for documentation release, software release and (if applicable) “promoting” to another library. Identify the internal documents where these library procedures are documented, formally or informally. **C3**

Q5 What is the program approach to establishing and controlling developmental baselines and test configurations? **C4**

Q6 Have the products to be placed under configuration management been identified? **C5**

Q7 What plans / practices assure consistent software identification across the programs functional disciplines? **C6**

Q8 How are software specification versions, and revisions for each release, formally identified within your SCM system? Is there a documented procedure for controlling the versions of software specifications? **C7**

Q9 Describe the process used to assure that each element of the software system is uniquely identified by name and number. **C8**

4 Quality Management and Product Control
4.7 Software Configuration Management
4.7.3 Configuration Audits

C1	Procedures and criteria are provided for a complete configuration audit including assigned responsibility. **Q1 Q2 Q3**	**Q1**	Who is responsible to perform and approve the configuration audits? **C1, C2**
C2	Software baseline audits are conducted. **Q1**	**Q2**	What are the criteria for a complete configuration audit? Where are the criteria documented? **C1**
		Q3	What procedure(s) are followed when performing software audits? **C1**

4 Quality Management and Product Control
4.7 Software Configuration Management
4.7.4 Configuration Control and Status Accounting

Criteria		Questions	
C1	A configuration control board (CCB) approach which addresses software is defined and integrated into the program change management review process. **Q1 Q4**	**Q1**	Does the Configuration Control Process include Configuration Control Boards with documented roles and responsibilities for each Board (list organizations represented)? Is there a hierarchy of CCBs? Explain this hierarchy. Is there a Software Configuration Control Board (SCCB) (list organizations represented)? How is it organized and what is its relationship with the CCB? **C1**
C2	Changes to baselines are controlled. **Q2 Q5**	**Q2**	Describe the procedures the program follows to control changes to configuration items? **C2**
C3	The software development process is used to manage and control the use of object code patches at all levels and assure that software patches are a rare exception and are always resolved with permanent changes. The use of patches are approved and documented by the configuration control board. **Q3 Q5**	**Q3**	Does your software development process allow the use of object code patches? How is this managed and controlled? How do you determine when a patch will be incorporated into the base line and when it will be deleted? How is this documented? **C3**
C4	Change requests and problem reports for all configuration items/units are initiated, recorded, reviewed, approved, and tracked. **Q4**	**Q4**	Does the Configuration management process include configuration status accounting? **C1 C4**
C5	Status accounting (status of configuration items/units) is recorded. **Q5**	**Q5**	How is status accounting achieved? Is the function automated? Describe the tools and process. **C2 C3 C5**
C6	The capabilities of the library and other tools support the status accounting functions. **Q6**	**Q6**	Explain how the capabilities of the library (or programming environment) and other tools support the status accounting functions. **C6**

4 Quality Management and Product Control
4.7 Software Configuration Management
4.7.5 Configuration Management Library and Tools

C1 A controlled access library system is in use which has procedures for software and documentation releases and for promotions to other libraries. **Q1**

C2 Engineering products are supported by the SCM library tools (e.g., requirements, documentation, source code-SDFs, design - SDLs) in an integrated manner. **Q1 Q2**

C3 Versions of the Software Engineering Environment tools are controlled and distributed among members of the development team consistent with the program plans. **Q3**

C4 SCM tools are proven and available for transition to the post-deployment support environment. **Q4**

Q1 How is (are) the software development library (libraries) organized? What specific products are supported by the SCM library tools, e.g., requirements, documentation, source code, design? **C1 C2**

Q2 What tools are used to implement SCM? Do the tools provide CM of the software, SDLs, documentation, and SDFs in an integrated manner? Which of these tools are automated? How are they integrated with the other software development tools such as in the software engineering environment? **C2**

Q3 How are the versions of the Software Engineering Environment tools (both core tools and extensions) controlled and distributed among members of the development team? **C3**

Q4 What assures that SCM tools are fully developed, proven and are available? What procedure is used to transition the tools to the post deployment support environment? **C4**

4 Quality Management and Product Control
4.8 Documentation
4.8.1 Identification, Production, and Control of Documentation

C1 There is a periodic update based on reviews held with the acquisition organization, user, and support organizations regarding the design information to be delivered. **Q1**

C2 All internal and deliverable documentation products are clearly identified, including draft documentation. **Q2**

C3 The documentation used to operate and maintain the software is developed and maintained consistently with the current software baseline. **Q3**

C4 Documentation is integrated into the engineering development process and the system, subsystem, and hardware, documentation. **Q4**

C5 Requirements for subcontracted software documentation are tied to the SEMS and SEMP. **Q5**

C6 Software documentation is integrated into the software development process and is supported by the S/SEE. **Q6**

C7 Automated tools support the generation of documentation. **Q7**

C8 The documentation is accessible from the S/SEE. **Q8**

Q1 Describe your process to define the detailed information to be included in the design documentation. How are the needs of acquisition organizations, users and supporters accommodated in this process? **C1**

Q2 Describe the process by which the necessary deliverable documentation is identified. How does incremental development affect documentation development? **C2**

Q3 How is documentation developed and maintained? What process(es) assures accuracy and completeness? **C3**

Q4 How is software documentation integrated into your software engineering development process? Describe the role of documentation in support of engineering activities? **C4**

Q5 Describe your requirements for subcontracted software documentation. Are these requirements tied to the SEMS and SEMP? **C5**

Q6 Is the key software documentation accessible from the Software Engineering Environment? **C6**

Q7 Identify any automated tools which support the generation of software documentation. Is the software technical requirements and development information accessible from the S/SEE? **C7**

Q8 Does the key software documentation exist in the Software Engineering Environment? What procedure assures that the S/SEE contains the current documentation baseline? **C8**

4 Quality Management and Product Control
4.8 Documentation
4.8.2 Technical Adequacy of Documentation

C1 Systems engineering decisions, development rationale, and test information are captured and retained in the documentation. **Q1**

C2 Standards exist for documenting test requirements for the software. **Q2**

C3 Internal standards or requirements for software documentation that are integrated with system, subsystem and hardware documentation are consistent with the requirements of the contract. **Q3**

C4 Documentation completion is a integral part of internal reviews to ascertain software development status and progress. Documentation requirements are part of the completion criteria for SEMS and SEMP. **Q4**

C5 Consistency and currency is maintained across software work products including the software plans, process descriptions, allocated requirements, software requirements, software design, code, test plans, and test procedures. **Q5**

Q1 Does the program have a process for capturing engineering decisions? Describe your approach for documenting design decisions, development rationale, and test information. **C1**

Q2 What standards do you use in documenting test requirements documents ? **C2**

Q3 Is the documentation approach integrated with the engineering activities? How does your software documentation integrate with system, subsystem and hardware documentation? **C3**

Q4 What role does documentation play in your internal reviews to ascertain software development status and progress? Do completion criteria for formal and informal reviews include appropriate documentation? **C4**

Q5 Is consistency maintained across software products from requirements through acceptance testing (i.e. traceability across software requirements, software plans, design, code and test)? What assures this is accomplished? **C5**

5 Organizational Resources and Program Support
5.1 Organizational Standards and Procedures
5.1.1 System and Software Development Processes

C1 The organization's systems and software development standards comprehensively describe the system and software development, their interfaces, and interdependencies. The standards also document the interfaces within and among the various system software and other disciplines. **Q1 Q2**

C2 The organizational standards provide a set of system and software engineering development models (e.g., waterfall, event driven) for selection and use by the program. The descriptions of these models are compatible with the organization's standard system and software development process(es). **Q3**

C3 The organization's system development and software development process(es) standards, where applied on the program, are compliant with applicable standards required by the RFP. **Q4**

C4 The organization's system development and software development process(es) standards are placed under configuration control. **Q5**

[If there is more than one standard process, answer the questions for the standard(s) applicable to the program.]

Q1 In your organization's system development and software development process(es) standards, how are activities and events described (e.g. inputs, outputs, readiness and completion criteria)? How are the relationships (sequencing, interfaces, and interdependencies) of the activities described? **C1**

Q2 What is covered in the descriptions of the interfaces within and between the various systems development and software development and other engineering development disciplines? For which disciplines are interfaces described? **C1**

Q3 Identify the system development and software development models (e.g. waterfall, event driven) and explain how these are defined in your standards. How is compatibility between the organization's standard system development and software development process maintained and ensured? **C2**

Q4 Which of your organizational development standards are compatible and compliant with the standards required by the program RFP. What is your approach for determining consistency? For those applicable standards required by the RFP, for which your standard process is not consistent, what is your approach for ensuring and supporting development processes that are compliant with program requirements. **C3**

Q5 Describe your approach for version control and controlling changes to the organization's standard system development and software development process(es). How do you know which version of the organization's standard is in use at a given time? How are changes to the standard assessed, incorporated within the standard, and incorporated by the program? **C4**

5 Organizational Resources and Program Support
5.1 Organizational Standards and Procedures
5.1.2 Tailoring

C1 A waiver procedure and tailoring guidelines and criteria are available to facilitate tailoring the organization’s standard systems development software development process(es) to meet specific program requirements and needs. **Q1 Q2**

C2 When organizational standards are applied directly to programs, a process exists to verify compliance with these standards. **Q3**

[If there is more than one standard process, answer the questions for the standard(s) applicable to the program.]

Q1 Describe any documented guidelines provided for tailoring organizational standards to specific program requirements. What specific program needs require tailoring on this program? How was the specific system development and software development model for this program selected? Given the systems and software development model for this program, how are the organization’s system and software development processes and procedures tailored to be compatible with and support the development model? **C1**

Q2 Describe the procedure for waiving compliance with the organization’s standard system development and software development process(es). How does it support application of the tailoring guidelines? Describe how the procedure provides flexibility for those cases where particular program needs require extensive tailoring. **C1**

Q3 How is compliance by program with the applicable organization’s standard process verified? How is compliance by program with the activity descriptions within the standard ensured or verified, taking into account approved deviations? How is the proper functioning of the interfaces within and between the various systems and software disciplines ensured or verified, taking into account approved deviations? **C2**

5 Organizational Resources and Program Support
5.1 Organizational Standards and Procedures
5.1.3 Capturing and Making Available Use Information

C1 Past use data for standard organizational and program processes is collected. These data include estimates and actuals, quality measurements, peer review/test coverage and efficiency, number and severity of defects found. These experience based data are made available to programs for planning and managing new programs. **Q1**

C2 A library of process-related documentation (e.g., program standards, measurement plans, process training materials) is maintained and made available to the program to support reuse of proven processes and interpretation of usage data. **Q2**

[If there is more than one standard process, answer the questions for the standard(s) applicable to the program.]

Q1 Explain how data from use of the organization's and programs' development processes and resulting products is collected and made accessible to the program for use in planning and managing its effort? In addition to the actual measurement data, what kind of related information is maintained to help the program understand and interpret the measurement data and assess it for reasonableness and applicability? **C1**

Q2 For the program, what kinds of process-related documentation is maintained and made available to support reuse of proven processes and interpretation of usage data? How are these documentation items catalogued for easy access? **C2**

5 Organizational Resources and Program Support
5.2 Facilities
5.2.1 Development Facilities

C1 A plan for establishing and maintaining the required system and software development facilities exists, and is consistent with the program's requirements, needs, usage estimates, and schedule. **Q1 Q2 Q3 Q4**

C2 Where required, integrated systems/software development environments are planned and acquired/developed to be in place to meet program need dates. **Q5**

C3 Planning for system and software development facilities includes support of all planned incremental development (blocks/builds), including regression testing. **Q6**

C4 For team developments, including primes, associates, and subcontractors, common and compatible development facilities are planned to ensure continuity, integrity, and supportability of the developed systems and software. **Q7**

Q1 Describe the software development facilities (host development computers, workstations, networks, memory systems, etc.) intended for the program in terms of quantity, location, availability date, capacity and response time. Describe the level of integration of the system/software development facilities (environments). **C1**

Q2 Describe the basis for determining that they will satisfy the program's requirements and needs (capabilities and capacities). **C1**

Q3 Describe when each facility will be available. What is your plan to ensure that the facilities will be available to meet the program's need dates? What is your fallback position should any of these facilities not be available in time for the program? **C1**

Q4 Describe how the development facilities will be maintained for the program. **C1**

Q5 For integrated software environments, describe which systems and software functions are integrated and how these integrated facilities will be planned and acquired or developed to meet program needs. **C2**

Q6 Describe how the software development facilities are planned and acquired to support incremental software development (blocks, builds). **C3**

Q7 For team developments, for example, primes, associates, and subcontractors describe how compatible development facilities are planned to ensure continuity, integrity, and supportability of the developed systems and software. **C4**

5 Organizational Resources and Program Support
5.2 Facilities
5.2.2 Specialized Facilities

C1 The specialized facilities that are needed for the program, such as special test stations and simulation labs, have been identified and meet the program's needs. A systematic process is used to ensure that all required facilities are identified, planned and acquired or developed to be in place to met the program need dates. **Q1**

C2 The availability of the specialized facilities is consistent with the program's requirements, needs, usage estimates, and schedule. **Q2 Q3**

C3 A plan to support these specialized facilities for the life-cycle duration of the program exists and is consistent with the program's resources and schedule. **Q4**

C4 Where required, integrated systems/software specialized facilities are planned and acquired/developed to be in place to meet program need dates. **Q5**

C5 Planning for system and software specialized facilities includes support of all planned incremental development (blocks/builds), including regression testing. **Q6**

C6 For team developments, including primes, associates, and subcontractors, compatible specialized facilities are planned to ensure continuity, integrity, and supportability of the developed systems and software. **Q7**

Q1 Describe the specialized testing facilities, simulation labs, and any other specialized facilities that will be used on this program. How do you ensure that all required facilities are identified, planned, and acquired or developed to be in place to meet program need dates. **C1**

Q2 Describe your plans for acquiring or developing the specialized facilities, if they are not already in place. What is your fallback position should any of these facilities not be available in time for the program? **C2**

Q3 Describe the process for scheduling the use of these facilities, and the number of shifts per day scheduled for each of them. How do you ensure that their availability will meet the needs of the program in terms of number of users and development schedule? **C2**

Q4 Describe how each one of these specialized facilities will be supported during the life of the program, in terms of user support and maintenance. Where is this support included in the program's allocated personnel and cost baselines? **C3**

Q5 For integrated specialized facilities, describe which systems and software functions are integrated and how these integrated facilities will be planned and acquired or developed to meet program needs. **C4**

Q6 Describe how the system and specialized software facilities are planned and acquired to support incremental software development (blocks, builds). **C5**

Q7 For team developments (for example, primes, associates, and subcontractors) describe how compatible specialized facilities are planned to ensure continuity, integrity, and supportability of the developed systems and software. **C6**

5 Organizational Resources and Program Support
5.3 Training
5.3.1 Training Plans

C1 A program training plan exists which identifies:
- the program's current and future technical, management, and skill needs
- how these needed skills will be developed (informal vehicles, formal courses that need to be developed, or procured from outside sources)
- the resources (e.g., trainers, materials, funding, time) needed to develop these skills
- the schedule for required training

Q1

C2 If the program's training plan relies on the organization, an organizational training plan exists which identifies:
- how organizational training needs are addressed (formal courses, etc.)
- the resources needed
- the schedule for conducting training
- relationship of organizational training to program training needs.

Q2

Q1 How are the program's software development training needs planned and implemented? Identify the skill needs that must be addressed. What training vehicles will be used to impart those skills? What resources are planned to develop those skills? Which training vehicles are provided by the program and which are provided by the organization? Does the schedule for required training meet the program need dates for skilled personnel? **C1**

Q2 Do the program's planned training needs rely on the organization for implementation? If so, describe what the organization's training plan covers. What organizational training needs are addressed? Which skill needs of the program are addressed? How does the training schedule reflect when those skills are needed by the program? What training vehicles will be used to impart those skills? What resources are planned to develop those skills? **C2**

5 Organizational Resources and Program Support
5.3 Training
5.3.2 Training Records and Effectiveness

C1 A procedure is established and used to determine whether individuals possess the knowledge and skills required to perform in their designated roles and to document training courses taken by individuals in their records. **Q1**

C2 Course effectiveness is evaluated, to help ensure that the training courses provide the required training. **Q2**

C3 The organization provides software training and motivation and incentives for personnel to take the training. **Q3**

Q1 Describe the process to determine whether individuals possess the software development knowledge and skills required to perform in their designated roles. Where skills are required to meet program needs, how are these skills developed? **C1**

Q2 Describe how the effectiveness of courses is evaluated. How are the results of the evaluations used to revise the training to better meet the specific needs they were intended to address? **C2**

Q3 What courses are provided by the organization and how are personnel encouraged to take the courses? **C3**

5 Organizational Resources and Program Support
5.3 Training
5.3.3 Training Requirements

C1 Technical and management skills training is provided for software development, including:
- software engineering development
- the programming languages (e.g., Ada)
- software engineering development environments (e.g., S/SEE)
- methods and tools
- software project management

Q1

C2 All new software development team members receive training in the program's software processes. **Q2**

C3 All new software project managers and program managers receive training or orientation, as appropriate, in such areas as software project management, program management, and system engineering. **Q3**

Q1 Identify the skills training courses typically provided for software development, in particular, software engineering development, the programming languages, and software project management development environments (e.g. S/SEE), methods and tools, and software project management. What program skill needs are addressed? Does the program depend on the organization to provide any of this training? **C1**

Q2 Is training provided in the program's system development and software development process and tools (e.g., S/SEE)? If such training is to be provided, what topics are covered? **C2**

Q3 What training and orientation is provided to new software engineers, project managers and program managers? Does the program depend on the organization to provide any of this training? **C3**

5 Organizational Resources and Program Support

5.4 Human Resources

5.4.1 Manpower Allocation Process

C1 A process exists for sizing the software development manpower requirement and for realistically allocating and distributing this manpower over the development phases. This process covers both technical and management manpower, is based on a documented model, and is calibrated on the basis of actual experience. The estimating and allocation process covers incremental software development. **Q1**

C2 The educational and training background of the proposed software development personnel is consistent with the program's skill needs. **Q2**

C3 The staff assigned to the subject program have the qualifications, technical skills, and experience in the application domains that are relevant to this program. **Q2 Q3 Q4 Q5 Q6 Q7**

C4 The lead software development engineers and managers assigned to the subject program has successfully demonstrated technical leadership and management skills on similar programs. **Q8 Q9**

Q1 Describe the manpower profile used to allocate the software development personnel over the total program development period, from requirements definition to subsystem/system testing, and explain the basis for this profile. **C1**

Q2 Describe the academic requirements/standards for software development engineers. **C2 C3**

Q3 What percentage of your software development personnel have scientific, engineering, mathematics, or computer science degrees? What percentage of the people assigned to the subject program will have this educational background? **C3**

Q4 Discuss the experience base and numbers of personnel required to accomplish the software development and related systems engineering and test on this program. **C3**

Q5 Identify the average years of relevant software development experience among your total software development staff. **C3**

Q6 What is the percentage of this experience acquired while with the current employer? **C3**

Q7 Describe the software development experience of your staff in terms of applications (domains) relevant to the subject program. **C3**

Q8 Describe the software management experience of your software management staff in terms of applications (domains) relevant to the subject program. **C4**

Q9 Describe instances from similar efforts where the lead software development engineers and managers assigned to this program have successfully demonstrated technical leadership and management skills. **C4**

5 Organizational Resources and Program Support
5.4 Human Resources
5.4.2 Manpower Availability and Retention

C1 The company's resources and assets are sufficient to accomplish the program's system/software development effort in parallel with the company's other ongoing and planned software development contracts and activities. Alternatively, management has a viable specific plan to acquire qualified personnel on a schedule consistent with the program's development plan. **Q1 Q2 Q3 Q4 Q5**

C2 A plan exists for ensuring the continued availability of qualified software development personnel throughout the life of the program. **Q6 Q7**

Q1 Identify all on-going and planned contracts which include software development, their magnitude, status and schedule. **C1**

Q2 Identify a composite company profile of software personnel working on all on-going and planned contracts. **C1**

Q3 Categorize these personnel by skills and experience and years of experience, including years with the company. **C1**

Q4 Demonstrate that the required numbers of personnel are either available within your organization to staff this program or how they will be acquired. **C1**

Q5 Describe the control and flexibility you have to assign and retain key people to this program. **C1**

Q6 Describe your process for assessing personnel stability, its relation to the application/function/subsystem, and your plans for maximizing it on the program. **C2**

Q7 Describe your process for tracking and managing personnel turn-over. **C2**

5 Organizational Resources and Program Support
5.5 Technology Assessment and Transition
5.5.1 Technology Transition Planning

C1 Program needs are analyzed to identify required capability areas that need or could benefit from new technology. **Q1**

C2 Responsibilities are assigned and resources allocated for monitoring, assessing, selecting, and adopting new technologies for the identified capability areas that need or could benefit from new technology. **Q2**

Q1 Describe your approach for determining the requirements/capability areas where new technologies are needed or would be most helpful. How do you factor in the on-going and planned program's needs? **C1**

Q2 Which group(s) or function(s) are responsible for monitoring, assessing, selecting, and adopting new technologies? Do these resources belong to the program only or to the organization? If for the organization, in what ways do their activities benefit the specific program? Which capability areas that need or could benefit from new technology are targeted for technology improvement? What kinds of expertise (e.g. technology change management, S/SEE, measurement) will be made available to help in assessing and transitioning technology changes? **C2**

5 Organizational Resources and Program Support
5.5 Technology Assessment and Transition
5.5.2 Technology Monitoring and Assessment

C1 Systematic efforts are made in the organization to identify and assess new technologies that might meet identified or anticipated needs. **Q1** **C2** Information on advanced technologies in use in the organization, which could benefit other programs, is disseminated. **Q2**	**Q1** How do you maintain awareness of commercially available technologies that might meet identified or anticipated needs? How do you maintain awareness of leading relevant technical work? What is your approach for gathering and reviewing documentation of experiences with using these technologies? **C1** **Q2** How do you maintain awareness of advanced technologies in use in the organization? What information on these technologies do you disseminate to benefit other programs? How do you disseminate this information? **C2**

5 Organizational Resources and Program Support
5.5 Technology Assessment and Transition
5.5.3 Technology Selection and Adoption

C1 Cost/benefit analyses are performed to determine the technology changes that will confer the highest potential benefits. **Q1**

C2 Where appropriate, pilot efforts are planned and conducted before a new or unproven technology is introduced into practice.
- pilot results reflecting on technically meeting the need, feasibility of adoption, and economics are collected, analyzed, and documented
- costs/benefits of broader use in the organization are estimated

Q2

C3 When a decision is made to introduce a new technology into practice, the applicable documented engineering development process(es) (e.g., program's, organization's) is updated to incorporate the new technology. **Q3**

Q1 How do you assess and evaluate a new technology to determine that it meets a technical requirement/need? Do you perform cost/benefit analyses on proposed technology changes? How do you use product and process data from the existing process in the cost/benefit analyses? What criteria do you use to determine the technology changes which will confer the highest potential benefits? **C1**

Q2 Describe your approach for piloting new or unproven technologies. What criteria do you use to determine whether a new or unproven technology should be piloted before incorporation? How do you plan for a pilot effort? How do you use the results of a pilot effort to assess the economics and feasibility of adopting the new technology? How are cost/benefits of broader use in the organization estimated? **C2**

Q3 When a decision is made to introduce a new technology into practice, how are the applicable documented process(es) (e.g. program's, organization's) updated to incorporate the new technology? **C3**

5 Organizational Resources and Program Support
5.6 Organizational Process Management
5.6.1 Process Planning and Coordination

C1	An organizational plan for improvement of system and software development process(es): - is based on the action plans resulting from assessments ofthe system and software development processes - identifies highest priority areas for improvement - indicates resources and assignments to develop the proces improvements - identifies applicable procedures - identifies how these improvements are incorporated into on-going and future programs **Q1 Q2**	**Q1**	How is the program plan for system and software development process improvement based on action plans resulting from process assessments? Which processes are covered in a process assessment? Are system processes included? How are findings from the assessment typically addressed (e.g. through action plans which identify the changes to be made)? What are the plan's highest priority areas for improvement? What are the program's priority areas for improvement and how are these addressed in the plan? **C1**
		Q2	Which activities are covered in the organizational plan for system and software development process improvement? Are group and individual responsibilities assigned and resources identified? Identify the procedures documented or referenced in your plan. How are improvements to be incorporated into on-going and future programs? **C1**
C2	The system and software process management activities of the organization are coordinated, in particular these activities: - defining and managing changes to the organization's system and software processes - collecting and maintaining data on use of the organization's system and software processes **Q3**	**Q3**	Which individual(s) or group(s) are responsible for coordinating the system development and software development process management activities of the organization? Who is responsible for managing changes to the organization's system development and software development processes? Who is responsible for collecting and maintaining data on use of the organization's system development and software development processes and making it available to other programs? How are these activities coordinated with the program? **C2**

5 Organizational Resources and Progam Support
5.6 Organizational Process Management
5.6.2 Improvement Process

C1 Specific opportunities for system and software development process improvement, addressing any area of system or software development process, are documented and can be proposed by anyone. **Q1**

C2 Systems and software development process improvement proposals are evaluated and decisions whether or not to implement them are made, based on expected benefits and relative priority. **Q2**

C3 When the decision is made to transfer a system or software development process improvement into a program, the improvement is implemented in a way that ensures:

- necessary resources to imple ment the improvement are determined and established
- the appropriate defined develop ment process(es) and training courses are updated
- consultation support is estab lished
- changes in development process performance are measured

Q3

C4 Managers and technical staff are in formed of the status and results of the organization's and programs' activities for system and software process development and improvement. **Q4**

Q1 By what mechanism(s) are specific opportunities for process improvement documented and submitted? Which employees can make of this mechanism(s)? What areas of the system and software development process can they address? **C1**

Q2 Describe how employee-identified and other proposed opportunities for process improvement are evaluated. What criteria are used to determine whether or not to implement a particular proposed improvement? How are benefits and priorities of a proposed improvements determined? Which group(s) or individuals(s) are assigned responsibility for evaluating and tracking these processes improvement proposals? **C2**

Q3 When the decision is made to transfer a system or software development process improvement into the program, what steps do you take to incorporate the improvement? What kinds of resources are assigned? How are the applicable document process(es) e.g., program's, organization's and training updated to incorporate the improvement? What training and consultation support do you typically plan to provide? How do you go about determining whether the change in process has improved technical performance and product and determine cost benefits? **C3**

Q4 What groups and functions are informed of the status and results of the organization's and programs' activities for system development and software development process improvement? How are they informed and how often? **C4**

5 Organizational Resources and Program Support
5.7 System/Software Engineering Environment
5.7.1 S/SEE Definition Process

C1 The S/SEE requirements definition process takes into account the needs of the program. **Q1 Q2 Q3**

C2 The S/SEE requirements definition process takes into account past usage of the S/SEE. **Q4 Q5**

Q1 Describe the process by which the S/SEE requirements are defined for the program. **C1**

Q2 How are the tools selected, and what is the involvement of program personnel in the selection? **C1**

Q3 How do you verify that the program's needs are met by the S/SEE? **C1**

Q4 How do you leverage on lessons learned from the use of this S/SEE on other programs? **C2**

Q5 Is there a mechanism for collecting S/SEE usage data and analyzing the effects of the S/SEE on software quality and productivity, and if so, are the results of this data collection used in the definition of the program's S/SEE? **C2**

5 Organizational Resources and Program Support
5.7 System/Software Engineering Environment
5.7.2 S/SEE Components

C1 The S/SEE components support the program's software engineering development and management requirements, functions, methodologies, and activities. **Q1 Q2 Q3**

C2 The S/SEE components are mature and well documented. New tools are determined through systematic evaluation to meet program needs. **Q4**

C3 The selected compiler has been benchmarked against the needs of the program in terms of specific domain/application requirements and functions. **Q5 Q6**

C4 The S/SEE components are selected to provide maximum commonality in support of an integrated development across team members, whenever prime/subcontractor teaming arrangements exist. **Q7**

Q1 Which software engineering development and software management requirements functions, methodologies, and activities are supported by the S/SEE and how? (Checklist in Table 5.7.2 can be used) **C1**

Q2 Describe how the hardware (configuration, hosts, targets, workstations, networks, disks, memory devices and systems, etc.) and associated operating systems support the needs of the program in terms of location, number of users, volume of computation, and compatibility with other contractors. **C1**

Q3 Describe how each tool in the S/SEE supports the software development process functions and methodologies selected for the program. **C1**

Q4 For each tool in the S/SEE, describe its maturity, the quality of its documentation, and how it will be supported during the program. Explain the rationale for selecting new (not yet matured) tools and how confidence is established in the ability of these new tools to meet program needs. **C2**

Q5 Describe your efforts in testing the selected implementation language compiler(s). **C3**

Q6 Describe your efforts in benchmarking the selected compiler relative to the specific program application/domain needs. Have any guidelines been generated for the use of the compiler on the program (special language features, Run Time System interface, , interface with other languages, etc.)? **C3**

Q7 Are all components of the S/SEE common across all members of the bidding team? Identify those that are not and provide a rationale for selecting them. **C4**

TABLE 5.7.2. TOOL CHECKLIST

1. General support tools
- Operating Systems
- Editors
- Document Production Tools

2. Specification Tools (Requirements and Design)
- Requirements Specification
- Requirements Analysis
- Design Specification
- Design Analysis
- Prototyping
- Traceability

3. Implementation Tools
- Automatic Code Generators
- Assemblers
- Compilers
- Linkers
- Code Analyzers

4. Testing Tools
- Debuggers
- Test Generators
- Data Generators
- Test Coverage Analyzers

5. Integration Tools
- Builders
- Loaders
- Performance Analyzers
- Throughput Analyzers
- Simulators
- Data Gatherers
- Data Reducers

6. Evolution Tools
- Configuration Management
- Change Analyzers
- Reverse Engineering

7. Management Tools
- Task Management
- Schedule Management
- Status Reporting
- Cost Estimation
- Size Estimation
- Metrics Analysis

5 Organizational Resources and Program Support
5.7 System/Software Engineering Environment
5.7.3 S/SEE Architecture

C1 The S/SEE is extendible, easy to use and well integrated. **Q1 Q2 Q3**

C2 The S/SEE memory utilization and throughput meet the program's needs. **Q4 Q5**

C3 The S/SEE security capabilities meet the program's needs. **Q6**

Q1 To what level are the different components of the S/SEE integrated? **C1**

Q2 Is there a common user interface to all the services provided by the S/SEE? **C1**

Q3 Describe the mechanism for adding new tools to the S/SEE. **C1**

Q4 What indicators are used to monitor the memory utilization and throughput of the S/SEE? **C2**

Q5 Describe the process for regenerating each of the software products that is generated with the S/SEE (a new software build, a new design specification, a new requirements specification, etc.) after an update has been made. Estimate the time duration based on software size similar to this program. **C2**

Q6 Describe the S/SEE security provisions and how they are used to manage unwanted intrusions and to protect information, consistent with the program's requirements. **C3**

5 Organizational Resources and Program Support
5.7 System/Software Engineering Environment
5.7.4 S/SEE Maintenance and User Support

Criteria		Questions	
C1	A process exists to ensure that S/SEE problems are identified, and corrected, and that changes to the S/SEE do not impact the program adversely. **Q1 Q2 Q3**	**Q1**	How do problems with the S/SEE get reported and corrected? **C1**
C2	A process exists to ensure that the program's S/SEE users are adequately trained and supported. **Q4 Q5**	**Q2**	How are changes to the S/SEE managed and controlled ? **C1**
		Q3	Describe the process of determining what portions of the S/SEE need to be recompiled when a component of the S/SEE is modified. When a change is made to the S/SEE, how is the potential impact on the program determined? **C1**
		Q4	Who has the responsibility to support the program's S/SEE users in their day to day use of the S/SEE? How is this support managed? **C2**
		Q5	Are there training courses on how to use the S/SEE? **C2**

5 Organizational Resources and Program Support
5.7 System/Software Engineering Environment
5.7.5 Deliverable S/SEE

Criteria		Questions	
C1	The S/SEE needed in the field to support all the deliverables has been identified. **Q1**	**Q1**	What are the S/SEE services that will be needed in the support phase of the program? **C1**
C2	The installation and support of the deliverable S/SEE has been planned. **Q2**	**Q2**	What resources will be needed for installing the deliverable S/SEE? Have they been planned? **C2**
C3	All restrictions on the use of the S/SEE or its components have been identified, and are consistent with the life cycle support requirements identified in the program's RFP. **Q3**	**Q3**	Describe your plans for supporting the deliverable S/SEE. **C3**
		Q4	Are there any restricted rights, licensing or other restrictions on delivering any components of the S/SEE? **C3**

6 Program Specific Technologies
6.1 Artificial Intelligence
6.1.1 AI Task Domain Analysis

Criteria		Questions	
C1	The offeror demonstrates an understand ing of the requirements they are propos ing to implement with AI technology. **Q1**	**Q1**	Characterize the problem that the AI implementation (system) addresses. **C1**
C2	The offeror demonstrates an understanding of the functional characteristics of the problem to which AI technology will be applied. **Q2**	**Q2**	What is the input/output behavior of the system? Can a sample dialog/script/ trace be provided? **C2**
C3	The offeror understands the time and space constraints under which the AI implementation will operate. This includes both development and operational constraints. **Q3**	**Q3**	What are the operational time and space constraints for the proposed system? **C3**
C4	The offeror is familiar with the state-of-the-art literature on similar systems. **Q4**	**Q4**	Describe the current academic, industrial, commercial, and government approaches to implementing similar systems. **C4**
C5	The offeror has past experience with AI solutions of the class they propose. **Q5**	**Q5**	Describe any previous experience with implementing AI solutions of the class proposed? **C5**

6 Program Specific Technologies
6.1 Artificial Intelligence
6.1.2 AI Tools and Technology

C1 The offeror demonstrates an understanding of the technology they are proposing, competing technologies, and justification of their chosen technology. **Q1 Q2 Q3**

C2 The offeror proposes and has the tools available to perform statistical analysis of the system behavior (e.g. number of rule firings, rule competitions, length of reasoning chains) or identifies plans for acquiring the tools. **Q4**

C3 The offeror has experience with any proposed COTS tools, or access to experience. **Q5 Q6**

C4 The offeror demonstrates an understanding of the canonical form of knowledge in the system (e.g. rules, frames, networks), and employs taxonomy for the types of knowledge used by the system. **Q7 Q8 Q9**

C5 A process exists for analysis of system behavior, and for verification and validation of the operational implementation. **Q10 Q11**

C6 The offeror has identified all of the tools necessary to develop the AI software. **Q12 Q13 Q15 Q16**

C7 All of the tools exist and have been successfully used by the offeror. **Q14 Q15 Q16**

Q1 What are competing approaches to implementing the AI solution? **C1**

Q2 Why is the approach proposed superior to competing approaches? **C1**

Q3 What are the limitations of the proposed approach? Are there special circumstances for which the proposed approach will not work? **C1**

Q4 Describe the reporting facilities of the proposed statistical analysis tools, the type of statistical data that the tools produce, and how that information will be used during the development. **C2**

Q5 Describe any previous experience with the proposed COTS products. In the absence of previous experience, provide the sources of expertise that will be applied to this program. **C3**

Q6 Why were the particular COTS products selected? What other COTS products exist for the task, and how are the selected products superior? **C3**

Q7 What is the form of knowledge in the proposed AI implementation? Is there a taxonomy of knowledge classes that the system uses? **C4**

Q8 Describe the knowledge engineering approach for the proposed AI implementation. Describe any previous experience in developing a knowledge base using this engineering approach. **C4**

Q9 What is the form of knowledge selected and how is it superior to other knowledge representation formalisms. **C4**

Q10 Describe the best and worst case performance expectations for the proposed AI implementation. **C5**

Q11 Describe methods for verification and validation of the operational implementation. **C5**

6.1.2	**AI Tools and Technology (cont.)**
	Q12 Identify the tools which comprise the proposed AI tool set. Describe the established tool set selection criteria. **C6** **Q13** Is this tool set complete relative to supporting the subject AI software development? If not, what tools are missing? How will these be acquired in time to support the development. **C6** **Q14** Describe any experience with the proposed tool set. Also describe the level of maturity of the proposed tool set. **C7** **Q15** Identify any limitations of the tool set and plans to work around these limitations. **C6 C7** **Q16** Identify any program specific enhancements planned for any of the tools. **C6 C7**

6 Program Specific Technologies
6.1 Artificial Intelligence
6.1.3 Specific AI Technology

C1 (Expert Systems) The offeror has access to experts from which the expertise can be extracted. **Q1 Q2**

C2 (Rule-Based System) The offeror demonstrates an understanding of why a rule-based approach is suitable for their problem (as opposed to an algorithmic approach). **Q3 Q4**

C3 (Off-line Training) The offeror can provide sufficient training data (including the source of the data and how well the data mirrors the distribution of operational input). **Q5 Q6**

C4 (Neural Networks) The offeror can identify the type of learning that the network will perform, and the equations that will be used by the connectionist units. **Q7**

C5 (Neural Networks) The offeror can identify the encoding of input and output data on feature vectors. **Q8**

C6 (Genetic Algorithms) The offeror has defined the evaluation procedures and how generations are constructed. **Q9**

C7 (Genetic Algorithms) The offeror has defined how the evaluation function relates to operational performance. **Q10**

C8 (Machine Learning Systems) The offeror demonstrates an understanding of when and what the system learns, and how the learned information is evaluated and used. **Q11**

C9 (Search Algorithms) The offeror demonstrates an understanding of the time and space requirements of the search procedure. **Q12**

C10 (State Space Search) The offeror demonstrates an understanding of the state space representation of the problem space. **Q13**

Q1 Describe the sources and methods used to acquire expert knowledge. **C1**

Q2 Describe methods for modifying the existing knowledge base as new knowledge becomes available. **C1**

Q3 Why is a rule-based approach better than an algorithmic approach? **C2**

Q4 What are the projected cost savings for using a rule-based approach over conventional programming techniques? **C2**

Q5 What is the source of training data? How does it correspond to input that the system will see operationally? How is it determined that the system will not overtrain on the training data? **C3**

Q6 Describe the procedure for generating training data. Describe the process for establishing the criteria for how well the system performs on testing data. Relate these criteria to operational usefulness. **C3**

Q7 Describe the type of connectionist network proposed, and the equations for weight modification and node firing. **C4**

Q8 How is input and output data encoded for use by the network? **C5**

Q9 What percentage of the population is used to construct the next generation? What mutation functions are used, and what percentage of a population is mutated? Describe the evaluation function, including how the genome is encoded. **C6**

Q10 How does the evaluation function relate to operational performance? (i.e. if the evaluation is .9, does that mean that the system will work 90% of the time?) **C7**

6.1.3 Specific AI Technology (cont.)

C11 (Heuristic Search) The offeror has demonstrated that their proposed search technique is admissible (always finds an answer), or optimal (always finds the best answer). **Q14**

C12 (Case-based Reasoning) The offeror demonstrates an understanding of the representation of a case, the indexing scheme for cases, how cases are compared, how cases are modified, and how cases are evaluated. **Q15**

C13 (Model-based Reasoning) The offeror demonstrates an understanding of the representation of a model, and how the model is used to evaluate assertions of system performance. **Q16**

C14 (Logic) The offeror demonstrates an understanding of the type of logic being proposed, and the proof procedure for formulas. **Q17**

C15 (Logic) The offeror demonstrates an understanding of the axioms used by the logic. **Q18**

C16 (Fuzzy Logic) The offeror demonstrates an understanding of the motivation and advantages of using fuzzy logic, and the overhead involved. **Q19**

Q11 What new knowledge does the system acquire? Describe the learning method? Is learning done off-line oroperationally? How does the system validate learned knowledge? **C8**

Q12 What are the time and space characteristics of the search procedure. **C9**

Q13 What is the representation of a node in the problem space? What are the operations/moves/arcs to move through the state space? Is the state space a tree or a graph? **C10**

Q14 Is the search algorithm admissible? Optimal? **C11**

Q15 What is the representation of a case? How are cases indexed, compared, modified, and evaluated? **C12**

Q16 How is the model represented? What methods are used to evaluate model performance? **C13**

Q17 What type of logic is being proposed, and what is the proof procedure used to prove formulas? **C14**

Q18 What are the axioms? What theorems have been proved about the axiom set? **C15**

Q19 What advantages does fuzzy logic have over conventional logic for the proposed solution? What overhead does the calculation of fuzzy values add? What method is used for propagating values? **C16**

6 Program Specific Technologies
6.1 Artificial Intelligence
6.1.4 AI Management Process

C1 The offeror has a documented process for the engineering management and development of AI software and systems. **Q1 Q2 Q3**

C2 The management process includes statusing and controlling mechanisms, with objective measures. **Q4 Q5**

C3 A process exists to establish schedule durations, milestones, and effort allocation for the AI development effort. **Q6 Q7**

C4 The offeror has identified any unique work package requirements driven by AI development management requirements. **Q8 Q9**

C5 The offeror has a process for estimating software size which is based on actual AI software development experience. **Q10 Q11**

C6 The AI technology necessary to develop the subject program exists and has been successfully applied by the offeror. **Q12 Q13**

Q1 Describe the process for managing the AI development effort. **C1**

Q2 What specific process steps are used to manage the AI software development process. What is the role of systems engineering? **C1**

Q3 Has the standard software development management process for AI been adapted? Describe these adaptations. **C1**

Q4 Describe the specific mechanisms proposed to status and control the AI software development over the system development period. **C2**

Q5 Describe how the AI development progress will be measured and assessed to include analysis, requirements definition, design, code implementation, integration and test. **C2**

Q6 How are schedule durations of the AI development and its impact on the total program schedule determined? **C3**

Q7 Describe the milestones, periods, and software effort allocated over the AI software development schedule. **C3**

Q8 Has the definition and implementation of work packages for the AI software development effort been modified compared to the standard process? If so, describe the work packages intended to be used to plan, define, control and status the development effort and why they have been modified. **C4**

Q9 Identify the milestone product completion criteria for the various phased products relative to work completion. **C4**

Q10 Describe the management approach to estimating the size of the AI software development effort? Identify any experience base used for this estimate. Describe how the estimating process reflects actual completed AI software development efforts? **C5**

6.1.4 AI Management Process (cont.)

Q11 Recognizing AI program size is not reflected in the empirically derived estimating models, how are estimates established for: **C5**

(1) Program Size

(2) Effort Required

(3) Development Schedules

(4) Distribution Of Effort Over The Schedule

(5) Cost

Q12 Identify any management concerns with the status of AI technology relative to developing software within the subject program baseline. **C6**

Q13 What are the specific strategies to manage the risk associated with the AI technology? **C6**

6 Program Specific Technologies
6.1 Artificial Intelligence
6.1.5 AI Development Process

C1 The offeror has an engineering process for AI software development which is documented and which has been successfully applied on past AI development programs. **Q1 Q2 Q3**

C2 The engineering process for AI software development includes:

(a) A systems engineering top-level architectural/design phase

(b) measurable milestones with completion criteria

(c) documentation of intermediate steps and final product design disclosure

(d) analysis, requirements definition, design, code, test and integration

(e) capturing and retaining the rationale behind AI design decisions.

Q1 Q4 Q5 Q6

C3 The AI software/system development process includes a comprehensive verification methodology and phase to validate that, e.g. the expert system, meets the specified performance. **Q7**

Q1 Describe the overall AI development approach. Identify each step in the process, and the products of each step. **C1 C2**

Q2 Identify any changes to the traditional software development process as a result of using AI technology. **C1**

Q3 Has the development process been verified through previous development, prototype development or IR&D? **C1**

Q4 Describe how the rationale behind the AI design decisions will be captured and retained. **C2**

Q5 Identify any special test levels that are unique to the AI applications. **C2**

Q6 Describe any special test facilities and resources required which are unique to AI applications. **C2**

Q7 How does the AI software/system development process validate the AI system performance? **C3**

6 Program Specific Technologies
6.1 Artificial Intelligence
6.1.6 Personnel Skills and Qualifications for AI

C1 The offeror has the necessary AI skills and experience to accomplish the AI software and system development. **Q1 Q2 Q3 Q4 Q5**

C2 The offeror's AI skills and experience are relevant to the subject program application. **Q6**

C3 A comprehensive AI training program exists which is sufficient to develop and maintain the skilled personnel for the subject program. **Q7 Q8 Q9 Q10 Q11**

C4 The offeror has the skilled and experienced personnel available to perform the development within the subject program baselines. **Q12 Q13 Q14**

Q1 Identify any staff AI development skills. Discuss all necessary AI skills to execute the subject AI development effort. **C1**

Q2 How were these AI skills acquired? **C1**

Q3 How is AI proficiency measured and evaluated in the various skills required? **C1**

Q4 Describe the corporate/division experience with actual application of AI. **C1**

Q5 Describe any experience with the defined development activities and phases? **C1**

Q6 Explain why the referenced AI experience is relevant and provides a basis for the subject program development. **C2**

Q7 Describe the training program followed to train personnel in AI. **C3**

Q8 Identify the total length of the training period and the subjects covered. **C3**

Q9 Does the training provide technical and management coverage ? Explain. **C3**

Q10 Is AI training required for all members of the staff? **C3**

Q11 How is proficiency developed following the initial training? **C3**

Q12 Demonstrate that sufficient AI trained and proficient personnel are available. How many are required throughout the development? **C4**

Q13 From where are these personnel coming? **C4**

Q14 What contingency provisions exist if enough personnel are not available? **C4**

6 Program Specific Technologies
6.1 Artificial Intelligence
6.1.7 AI Capability Demonstrations and Risk Management

C1 The offeror has planned specific demonstrations to establish AI technology and related tools exist in a form sufficiently mature and dependable to perform the subject program AI software development. **Q1 Q2**

C2 The offeror has documented risk management methods to assure a successful development effort within the subject program baselines. **Q3 Q4 Q5 Q6 Q7**

Q1 Describe planned demonstrations to establish the AI tools, capability and approach to this development program. **C1**

Q2 What are the schedules and criteria for these demonstrations? **C1**

Q3 Describe any short falls or deficiencies seen in the AI technology base to support the development effort. **C2**

Q4 What back-ups, provisions, and contingency plans exist to compensate for these short falls? **C2**

Q5 Recognizing AI as an evolving technology, what risks exist in developing the subject program using AI within program constraints? Identify specific technical and management risks. **C2**

Q6 Describe any defined specific risk management provisions planned for use. **C2**

Q7 Describe the criteria used to exercise each risk management provision. **C2**

6 Program Specific Technologies
6.2 Safety Critical Digital Systems (SCDS)
6.2.1 Safety Critical Program Management

C1 An approach is defined to account for changes to Government Furnished Equipment and Contractor Furnished Equipment safety critical items which impact development efforts. **Q1**	**Q1** Describe the proposed approach to accounting for changes to Government Furnished Equipment and Contractor Furnished Equipment safety critical items which impact the development efforts. **C1**
C2 Interfaces and management agreements are proposed to facilitate the communication and interaction of the development organization and other safety critical developers/maintainers, other non-safety critical developer/ maintainers, the subsystem IPT lead management functions, and the organization (IPT) responsible for management at the system level, e.g. aircraft level? **Q2**	**Q2** What formal or informal agreements exist between the development organization and other safety critical developers/ maintainers, other non-safety critical developer/ maintainers, the subsystem IPT lead management functions, and the organization (IPT) responsible for management at the system level, e.g. aircraft level? **C2**
C3 The safety certification process is defined in the SDP or other appropriate vehicles. **Q3**	**Q3** Is the safety certification process defined in the SDP? If not, describe the vehicle used to define the certification process. **C3**
C4 The schedule estimation and definition system proposed accommodates the safety critical development activities defined by the offeror. **Q4**	**Q4** How is it assured that the schedule will accommodate all the safety critical activities required? **C4**

6 Program Specific Technologies
6.2 Safety Critical Digital Systems (SCDS)
6.2.2 Safety Critical Systems Engineering

C 1 A process is defined to accomplish or update system wide safety and hazard analysis. This process includes criteria for defining and identifying safety critical elements including Safety Critical Subsystems, Components, and Software. **Q1 Q2 Q3 Q4 Q10**

C 2 A process exists to assess system safety program requirement tasks for applicability and incorporation into organizational standards and procedures. **Q5**

C 3 A process exists to incorporate the results of system wide hazard analyses into specific system safety requirements for software development including system, subsystem, and software specifications. **Q6 Q11**

C4 A mechanism is defined to identify system and software functions which are essential to safe operation. Criteria for establishing critical elements and associated testing are defined. **Q7**

C5 A process is used to ensure that design changes account for and do not violate existing safety analyses and trade studies. **Q8**

C6 A process exists which defines how system/subsystem component qualification/re-qualification with actual hardware and the latest operational version release of software is performed. A process is defined to qualify/re-qualify software associated with hardware modifications which affect software performance/ timing/sizing. **Q9**

C7 The process for performing and updating fault trees (IFAS), Failure Mode Effects Analysis and Failure Modes Effects and Criticality Analysis is defined. Mature tools are available and experience using these tools is demonstrated. **Q10**

C8 A system/subsystem architectural analysis process is used to verify the architecture meets the identified system level safety requirements. **Q12**

Q1 Provide the baseline definitions of Safety Critical Subsystems, Components, and Software. **C1**

Q2 Describe the approach and processes to accomplish a system wide safety analysis. Describe the process and criteria for identifying safety critical subsystems, components, hardware and software. What criteria are used to identify safety critical functions at the system and subsystem level? Describe the approach to updating and maintaining any existing safety hazard analyses. **C1**

Q3 Describe the software hazard analysis process performed on safety critical software. How are identified hazards associated with software assessed for hazard severity and probability of occurrence? **C1**

Q4 Describe the process to ensure that all flight critical/safety critical functions and systems have been identified. **C1**

Q 5 How are the software safety tasks assessed for applicability? Which tasks are included in the organizational standards and procedures? Identify any differences between these standards and procedures and the system safety program requirements. **C2**

Q 6 How are the results of system wide hazard analyses transformed into specific system safety requirements for software development? **C3**

Q7 Describe how the functions which are essential to safe operation are determined. What criteria are used to determine the flight critical areas and the associated testing required? What criteria establishes a function as being critical? **C4**

Q8 Describe the mechanisms established which will ensure that design changes account for and do not violate existing safety analyses and trade studies. Describe the approach for updating these analyses and trade studies. **C5**

6.2.2 Safety Critical Systems Engineering (cont.)

C9 A process is proposed which assures the autonomy and integrity of the safety critical digital systems. **Q13**	**Q9** Discuss the process to perform system/ subsystem component qualification/ requalification with actual hardware and the latest operational version release of software. Discuss how hardware requalification will be addressed for all modifications planned. How will software qualification/requali-fication be managed for an associated hardware modification which affects software performance/ timing/ sizing? **C6** **Q10** Have the safety critical hardware and software components been identified? Describe the use of the safety analysis process and supporting tools such as fault trees (IFAS), FMEAs, and FMECAs in identifying safety critical hardware and software components at the system and subsystem level. Describe any experience in using the tools on previous developments. From these, has a subsystem safety analysis, Failure Mode Effects Analysis and a FMECA been performed? Is it kept current with each modification to assess safety critical aspects/risks? **C1, C7** **Q11** Describe the methodology to be used to identify specific safety requirements to be integrated into the software development specification. **C3** **Q12** What specific process and procedures are used to verify that the top-level system / subsystem architecture meets the system level safety critical requirements, e.g. fault identification, fault tolerance? How does this process ensure flight critical systems are designed with the necessary redundancy management to accommodate fault tolerant reliability specific requirements? **C8** **Q13** Describe how the top-level design approach assures the autonomy and integrity of safety critical digital systems. How does this process identify safety critical component interconnectivity as itrelates to the identification and control of error propagation through the system? How does this process prevent the contamination of safety critical systems and components by non-safety critical systems? **C9**

6 Program Specific Technologies
6.2 Safety Critical Digital Systems (SCDS)
6.2.3 Safety Critical Software Engineering

C1 A process is proposed and defined to translate and trace safety specific system requirements into the software requirements baseline. **Q1**

C2 The identified top-level design process defines the method to verify the design against the established safety requirements, products and completion criteria. **Q2**

C3 A process exists to evaluate and analyze the software design to hazardous conditions. **Q3**

C4 A process exists which continues the identification of safety critical elements down to CSCs and CSUs. **Q4**

C5 A process is proposed which assures the autonomy and integrity of the structure and interfaces between safety critical elements and non-safety critical elements. (CIs, CSCIs, CSCs, and CSUs). **Q5 Q6**

C6 A process exists to trace safety critical parameters within the code to the functions which modify them or to the functions which use these safety critical parameters. **Q7**

C7 If specific safety critical coding standards are used, these standards are defined and verified as to compliance. **Q8**

Q1 Describe the specific process to translate and trace safety specific system requirements into the software requirements baseline. Describe how traceability of these software safety critical requirements back to their higher level system and subsystem level safety critical requirements is maintained. **C1**

Q2 What is the product(s) of the top level design activity? What are the criteria for completion of the top level design activity? Describe the process to assess whether the software design implements (satisfies) the established safety requirements. **C2**

Q3 Describe the process to evaluate and analyze software designs (top level and detailed level) for hazardous conditions. **C3**

Q4 Describe the process to identify safety critical CSCs and CSUs. **C4**

Q5 Describe the processes and procedures to ensure autonomy of structure and interfaces between safety critical components/ CIs and non safety critical components/ CIs. **C5**

Q6 What process is used to insure that design changes to non-safety critical CSCs and CSUs do not adversely impact safety critical CSCs and CSUs? **C5**

Q7 Describe the approach to verifying that safety critical parameters are properly traced within the code to the functions that modify them or to the functions which use these safety critical parameters? Does this approach allow impact analyses performance to determine if modifications made to any part of the system will have some affect on critical parameters contained in safety critical software? Does this approach facilitate mapping safety related message paths with external interfaces? **C6**

Q8 What internal standards and procedures define the safety critical coding standards to be applied? Which organization verifies compliance to the established coding standards? **C7**

6 Program Specific Technologies
6.2 Safety Critical Digital Systems (SCDS)
6.2.4 Safety Critical Software Test and Integration

C1 Regression test procedures are defined from CSU to CSCI/CSCI integration including the use of a core test process if planned. **Q1 Q10**

C2 Test coverage procedures are defined for unit test including the process to execute all software instructions and branches during unit testing. **Q2**

C3 Test coverage procedures are defined to ensure that all safety critical software is tested at and beyond the systems limits, with abnormal/erroneous conditions, as well as all transition points (e.g. mode to mode). **Q3 Q4 Q5**

C4 Integration and test procedures exist to perform timing and sizing analysis verification. **Q6**

C5 Test cases, descriptions, procedures and reports are maintained/updated for each level of test from unit test to CSCI test. **Q7 Q8 Q9**

C6 A process exists for determining the level of test for safety critical components. **Q10**

C7 The software test planning process incorporates an analysis of whether the use of fault injection is warranted. **Q11**

C8 A process exist to define the required target digital processor and other system hardware in the successive build up to the software integration and test. Alternative plans are defined if the required integration test hardware is unavailable. **Q12**

Q1 Describe the process for regression testing from CSU to CSCI/CSCI integration? Describe how core test cases are identified at all levels of testing for critical functions within the software/hardware. What process ensures that the core tests include all portions of the software testing which must be executed to ensure that operation of critical functions are safe and work as intended? **C1**

Q2 Describe the process to ensure that all of the software instructions are executed during unit test? Describe the process to ensure that all branches within the unit are tested during unit test. **C2**

Q3 What are the processes and procedures to test at and beyond the limits (in-bounds, out-of-bounds) as well as at all transition points? **C3**

Q4 What specific process is used to verify/assess the adequacy of the software test coverage? **C3**

Q5 Describe the process for performing abnormal/erroneous condition testing at each level of identified testing. **C3**

Q6 Describe the process for performing timing and sizing analysis verification. Is this process contained in standard integration and test procedures? **C4**

Q7 Describe how unit test cases are maintained/updated during unit test. How are these test cases used for regression testing? Are these unit test cases maintained in the SDF? **C5**

Q8 Which vehicles/procedures/methods are used to maintain test software descriptions and procedures for re-test? Are these procedures maintained in the SDF? **C5**

Q9 Which organization maintains the test results for each level of test? **C5**

6.2.4 Safety Critical Software Test and Integration (cont.)

Q10 Describe the process for determining the required level of test and re-test for safety critical components. For example, if an error requiring a code change is discovered in subsystem testing, describe the levels and completeness of the re-test starting at the unit level and progressing to higher levels. **C1, C6**

Q11 How is fault injection used in the validation and verification of safety critical requirements performance? What process is used to decide if fault injectin is appropriate? **C7**

Q12 Describe the use of target digital processor and other system hardware in the successive build up of the software integration and test. In the absence of actual hardware, how are software functions which are dependent on hardware interfaces tested? **C8**

6 Program Specific Technologies
6.2 Safety Critical Digital Systems (SCDS)
6.2.5 Safety Critical Subsystem/System Test and Integration

C1 A process exists for including established criteria to control the test article in regards to discrepancy resolution incorporation into system level test phases such as DT&E flight test. **Q1**

C2 Processes and procedures exist to define the depth and completeness of the re-test effort for each level of subsystem integration and system level testing. **Q2**

C3 A process exists to maintain configuration control of the test environment including hardware and software during hardware/software testing as well as higher levels of subsystem and system testing. **Q3**

C4 Integration and test procedures exist to perform timing and sizing analysis verification at the subsystem and system level. The timing and sizing analysis verification results are maintained. **Q4**

C5 A process is defined to perform abnormal/erroneous condition testing at the subsystem and system level. **Q5**

C6 Regression test procedures are defined for hardware/software integration, subsystem test and integration and system test including the use of a core test process if planned. **Q6**

C7 The software development/generation process facilitates updates to safety critical systems without the use of object code patches. The process for accomplishing this is defined and ensures that patches are not promulgated into safety critical software. **Q7**

C8 Test cases, descriptions, procedures and reports are maintained/updated for each level to test during subsystem and system test. Processes exist which define how the test cases, descriptions, procedures and reports are for regression testing. **Q8**

Q1 What criteria are established to control the test article in regards to discrepancy resolution incorporation during system level test phases such as DT&E flight test. **C1**

Q2 Describe plans for re-test management. Describe the process and practices which define the depth and completeness of the re-test effort for each level of subsystem integration and system level testing. **C2**

Q3 Describe how the configuration of the test environment including hardware and software is maintained and controlled for hardware/software testing as well as higher levels of subsystem and system integratin and test. **C3**

Q4 Describe the process to perform timing and sizing analysis verification at the subsystem and system level. How are the results of this verification maintained? **C4**

Q5 Describe the process for performing abnormal/erroneous condition testing at the subsystem and system level. Is Failure Modes Evaluation Testing (FMET) used? **C5**

Q6 What plans exist for regression testing from subsystem to system level? Describe how core test cases are identified at all levels of testing for critical functions within the system. What process ensures that the core tests include all portions of the system testing which must be executed to ensure that operation of critical functions are safe and work as intended? **C6**

Q7 How does the software development process preclude the use of object code patches at any level relative to SCDS software? How are the changes to the software during development and test accomplished to assure patches are not promulgated into flight and safety critical software? **C7**

6.2.5 Safety Critical Subsystem/System Test and Integration (cont.)

C9 An approach to ensure adequate subsystem development team involvement and support of system level integration and test activities for SCDS is defined. This approach defines the subsystem development team involvement in the following levels of system integration and test:

a) System Integration Laboratory Testing

b) Dynamic Simulation Testing

c) On-System ground test

d) Flight tests

Q9

C10 A procedure exists and is used to analyze and determine the level of fidelity required at each level of subsystem and system test. **Q10**

Q8 Describe how subsystem/system test descriptions, test procedures and test cases are maintained during subsystem and system test. How are these procedures used for regression testing? **C8**

Q9 Describe the approach to ensure adequate subsystem development team involvement and support of system level integration and test activities for SCDS covering the following levels: **C9**

1) System Integration Laboratory Testing
   - Static environment testing
   - Interface compatibility
   - Communication/timing
   - Operability
2) Dynamic Simulation Testing
   - Operational (actual) hardware/software.
   - High fidelity environmental/simulation models
   - FMET
   - Performance/operational confidence tests
   - Interface compatibility validation
3) On-System ground test
   - System operational/compatibility/connectvity/integration tests
   - Ground vibration and other ground safety checks
   - EMC validation
4) Flight tests
   - Controlled envelope expansion.
   - Test start/objectives/criteria defined
   - Test levels/quantities/coverage analysis process

C9

Q10 Describe the process to analyze and determine the level of fidelity required at each level of subsystem and system testing. Describe how the proposed facilities satisfy the established fidelity requirements. **C10**

6 Program Specific Technologies
6.2 Safety Critical Digital Systems (SCDS)
6.2.6 Safety Critical Personnel Resources

C1 The application specific personnel resources needed to support the safety critical design, development, integration and test effort are available or a plan is defined to acquire the needed resources. **Q1**

C2 The application specific personnel resources available possess the relevant experience, academic qualifications and programming language skills and experience required for the subject program. **Q2 Q3 Q4**

Q1 Are the application specific personnel resources needed to support the safety critical design, development, integration and test effort available? If not, describe plans to acquire the needed resources. Address the following combinations and types depending on the application:

1) Functional Engineers including safety engineering
2) Systems engineers with specific functional and integration experience along with a complete understanding of SCDS architecture and integration criticality.
3) A strong understanding of how the software functions integrate with the subsystem/system and the software development process which must be followed to minimize for changes made to SCDSs.

C1

Q2 Identify the average years of relevant safety critical system development experience among the safety critical development staff and contractors. **C2**

Q3 What are the academic requirements/ standards for software/system safety engineers to be used in the safety critical software development? **C2**

Q4 Describe any experience with actual application of the selected programming language to SCDS applications. How is this experience relevant for this development? **C2**

6 Program Specific Technologies
6.3 Complex Hardware Development
6.3.1 Hardware Management

C1 A process for managing complex integrated circuit (CIC) development and procurement is described. This process includes the following:

(1) Workload estimates and budgets

(2) Development schedules and relation to specific work packages

(3) Methods for tracking progress of individual work assignments

(4) Work assignment prioritization

(5) Budget and schedule impacts

Management tools used in the requirements flowdown and design process are identified. **Q1 Q2**

C2 A documented process for managing the flowdown of CIC hardware requirements to individual ICs is described. **Q3 Q4**

C3 A method to coordinate hardware and software designs and resolve conflicts is implemented. **Q5**

Q1 Describe the basic process used to manage complex integrated circuit (CIC) hardware development. **C1**

Q2 How is the workload estimated and budgeted? How are overall schedules developed and how do they relate to specific work packages assigned to individuals? How are work assignments prioritized and what system is used to track progress of individual work assignments? How are budget and schedule impacts identified “from the bottom up” via these detail assessments? **C1**

Q3 Identify any tools used to assist in the management of the CIC hardware requirements flowdown and design process. These may include computer-based COTS packages, in-house systems, or non-automated accountability systems. **C2**

Q4 What is the organizational structure for managing the flowdown of CIC hardware requirements to the level of individual complex ICs? **C2**

Q5 How will CIC hardware designers coordinate their designs with those of software designers via this organizational structure? How are conflicting concepts in the (hardware and software) designs surfaced and resolved? **C3**

6 Program Specific Technologies
6.3 Complex Hardware Development
6.3.2 Hardware Subcontractor Management

C1 Management standards are imposed on subcontractors which require status reporting compatible with the in-house management information system. **Q1**	**Q1** What management standards for CIC hardware development are imposed on subcontractors? What information is required from subcontractors to report status? In what form is this information required to be submitted (i.e., compatible with a particular COTS package)? How is this information used with in-house management information to assess status of the project? How is this subcontractor information made visible/ available to the government? **C1**

6 Program Specific Technologies
6.3 Complex Hardware Development
6.3.3 Hardware Design and Test

C1 A documented process is described to allocate design requirements to hardware and software and to identify specific complex integrated circuit types. **Q1**

C2 Design validation is accomplished prior to release to fabrication. **Q2 Q3**

C3 In-house standards for acceptance of CIC hardware have been established. **Q4**

C4 CIC documentation standards are established consistent with the need for long term support and reprocurement by the program. **Q5**

Q1 How are design requirements, once allocated to hardware and software in general, flowed down to an identified need for a particular complex IC (e.g., an ASIC or gate array)? Describe where this flowdown is documented, including documentation of the specific, detailed requirements to be met by the identified complex circuit. How is this flowdown process and documentation of the results made visible/available to the government? **C1**

Q2 Describe the process for design validation of complex integrated circuits (e.g., ASICs, complex gate arrays, VHSIC) prior to their release to fabrication. This description should include identification of design tools and methodologies, hardware and/or software simulators, development of test vectors for on-board BIT circuits, and any standards to be met prior to release of the IC design. **C2**

Q3 Describe the process to insure that vendor supplied chips are designed properly with regard to tolerance build-up, timing variability and other similar concerns. How are these design requirements documented? **C2**

Q4 What standards exist for in-house acceptance of CIC hardware? **C3**

Q5 Describe the documentation standards for complex integrated circuits developed for this project. How is transportability of this design information to the required VHDL "language" assured? How will the documentation produced through these standards be sufficient to permit future design changes in the IC's functionality by other than the original developer? **C4**

6 Program Specific Technologies
6.4 Database Management
6.4.1 Need for Database

C1 The offeror's processes and procedures address initial tradeoff issues relating to database usage as an implementing technology. **Q1 Q2 Q3 Q4**

Q1 What issues have been identified that indicate a need to use a database? **C1**

Q2 What attributes of databases would effectively address the issues identified? **C1**

Q3 In terms of life-cycle support, how does the support of a database in the operational mode compare to that of the alternative solution? **C1**

Q4 For a distributed or heterogeneous environment, what are the critical technical issues in database technology, and how will they be addressed? **C1**

6 Program Specific Technologies
6.4 Database Management
6.4.2 Database Tools

C1 The offeror demonstrates an understanding of how to identify the best database methodology to satisfy the system requirements. **Q1 Q2 Q3 Q4 Q5 Q**

C2 The offeror demonstrates an understanding of how to identify the tools required and to document the selection rationale. **Q7 Q8 Q9 Q10 Q11 Q12 Q13 Q14**

Q1 What is the selected database methodology? **C1**

Q2 What are the competing approaches for satisfying the system requirements? **C1**

Q3 What are the advantages of the selected approach? **C1**

Q4 What are the limitations of the selected approach? **C1**

Q5 How does the defined database methodology integrate with the proposed software development methodology? **C1**

Q6 Describe any experience with this database methodology? **C1**

Q7 What are the competing COTS products? **C2**

Q8 What selected tools are available, and how will each one support this system (program requirements, hardware, operating systems, user interface, interface with the selected programming language)? **C2**

Q9 How do the tools selected support the selected methodology? **C2**

Q10 Describe the commercial vendor's place in the market relative to other database products. **C2**

Q11 Define the support and training available from the vendor. **C2**

Q12 For the products selected, show the vendor's compliance to any required standards , as well as, the commitment to the evolution of these standards. **C2**

Q13 Describe the levels of security provided by the selected database tools, as needed. **C2**

Q14 Describe any experience with the selected database tools? What further training will personnel need to use the selected tools and keep current with the upgrades? **C2**

6 Program Specific Technologies
6.4 Database Management
6.4.3 Database Development

C1 The database development process and procedures are defined in internal development standards and procedures and include internal reviews, walkthroughs, statusing, testing and discrepancy resolution. **Q1 Q2**

C2 The database development processes and procedures are compatible with the selected database methodology. **Q3 Q4 Q5 Q6**

Q 1 Describe the database development processes and phases with respect to the software development. Do these phases provide a realistic schedule for the database development? **C1**

Q2 Describe the internal review process of the database development in terms of software and requirement reviews. Who is involved? Who is ultimately responsible? How are discrepancies resolved? Define the progression of testing through internal reviews, walkthroughs, and software inspection. **C1**

Q3 Describe how the selected database development process is compatible with the selected database methodology. **C2**

Q4 Describe the data modeling to be used. How does the data modeling support the database development? **C2**

Q5 How is the database methodology used throughout the database development, database populating, and database processing? **C2**

Q6 How is data integrity enforced during development? **C2**

6 Program Specific Technologies
6.4 Database Management
6.4.4 Database Quality Assurance

C1 A process is defined for verification and validation of the database system. **Q1 Q2 Q3 Q4 Q5**

Q1 Describe the methods for verification and validation of the database system. **C1**

Q2 Describe how the software releases and database releases are configured with one another. **C1**

Q3 How is the interface between the software and DBMS tested? **C1**

Q4 How is a database release version documented? Who controls releases? **C1**

Q5 How is the integrity of the data ensured and controlled? **C1**

6 Program Specific Technologies
6.4 Database Management
6.4.5 Personnel Skills and Qualifications for Database

C 1 The offeror has the necessary database skills and experience to accomplish the database software and system development. **Q1 Q2 Q3 Q4 Q5**

C2 The offeror's database skills and experience are relevant to the subject program application. **Q6**

C3 A comprehensive database training program exists which is sufficient to develop and maintain the skilled personnel for the subject program. **Q7 Q8 Q9 Q10 Q11**

C4 The offeror has the skilled and experienced personnel available to perform the development within the subject program baselines. **Q12 Q13 Q14**

Q1 Identify any staff database development skills. Discuss all necessary database skills to execute the subject database development effort. **C1**

Q2 How were these skills acquired? **C1**

Q3 How is database proficiency measured and evaluated in the various skills required? **C1**

Q4 Describe the corporate/division experience with actual application of databases to MCCR applications. **C1**

Q5 Describe any experience with the defined development activities and phases? Explain. **C1**

Q6 Explain why the referenced database experience is relevant and provides a basis to do the subject program development. **C2**

Q7 Describe the staff training program followed to train database development skills. **C3**

Q8 Identify the total length of the training period and the subjects covered. **C3**

Q9 Does this training provide technical and management coverage? Explain. **C3**

Q10 Is database development training required for all members of the staff? **C3**

Q11 How is proficiency developed following the initial training? **C3**

Q12 Demonstrate that sufficient database development trained and proficient personnel are available. How many are required throughout the development? **C4**

Q13 From where are these personnel coming? **C4**

Q14 What contingency provisions exist if enough personnel are not available? **C4**

ATTACHMENT 1. ACRONYMS, DEFINITIONS, AND REFERENCES

Section A. Acronyms

AFFARS	Air Force Federal Acquisition Regulation Supplement
AFMC	Air Force Materiel Command
AI	artificial intelligence
BAFO	best and final offer
CBD	Commerce Business Daily
CC	Critical Capability
CCA	Critical Capability Area
CDRL	Contract Data Requirements List
CMM	Capability Maturity Model
COTS	commercial off-the-shelf
CR	clarification request
CSCI	computer software configuration item
Dem/Val	Demonstration/Validation
DPRO	Defense Plant Representative Office
DR	deficiency report
EMD	Engineering and Manufacturing Development
FA	Functional Area
GNTO	general notice to offerors
IIV&V	internal independent verification and validation
ITO	instructions to offerors
KSLOC	thousands of source lines of code
MCCR	mission critical computer resources
OPR	office of primary responsibility
PAT	Process Action Team
PCO	procuring contracting officer
PRAG	Performance Risk Assessment Group
RFP	request for proposal
SCE	Software Capability Evaluation
SCM	software configuration management
SDCCR	Software Development Capability/Capacity Review
SDCE	Software Development Capability Evaluation
SDP	Software Development Plan
SEDS	Systems Engineering Detailed Schedule
SEI	Software Engineering Institute
SEMP	Systems Engineering Management Plan
SEMS	Systems Engineering Master Schedule
SOW	statement of work
SPO	system program office
SQA	software quality assurance
SS	source selection
SSA	Source Selection Authority
SSAC	Source Selection Advisory Council
SSEB	Source Selection Evaluation Board
S/SEE	System/Software Engineering Environment
SSEG	source selection evaluation guide

* All source selection descriptions in this pamphlet and the definitions in this glossary are consistent with, and in many cases are direct quotes or paraphrases from: AFAC 92-33, Air Force FAR Supplement Appendix AA, “Formal Source Selection for Major Acquisitions.”

Section B. Definitions

Acceptable
See: Scores and Ratings

AFMC SDCE Office of Primary Responsibility
This office is responsible to lead AFMC in the application and support of the SDCE method. This includes coordinating with HQ AFMC/EN and with Center OPRs for the SDCE, developing SDCE training materials, and improving/updating the SDCE method. Metrics gathered in using the SDCE will be provided to the AFMC OPR through the Center OPRs. The AFMC SDCE OPR is ASC/EN(CR).

Area
See: Specific Criteria

Assessment Criteria
See: Criteria, Assessment

Associate Contractor
See: Contractor

Bidder
See: Contractor

Blue
See: Scores and Ratings

Center SDCE OPR
Each AFMC center that does source selections will have an SDCE OPR to lead and support the application of the SDCE method at their that Center. This includes providing consultation and advice to program offices in applying the SDCE. The Center OPR will provide and refine SDCE training to meet the Center's needs. The center OPR will collect metrics on applying the SDCE from the center program offices using the SDCE and provide summary metrics to the AFMC OPR. These metrics will be used to improve the SDCE method.

Clarification Request
If data provided in the proposal is inadequate for evaluation or contradictory statements are found, a clarification request should be developed. Two categories of clarification request exist. Significant CRs specifically identify the aspect of the offeror's proposal for which clarification is required. Resolution of significant CRs requires that discussions with offerors be opened. Minor CRs are for the purpose of eliminating minor irregularities or apparent clerical mistakes. Minor CRs do not give the offeror an opportunity to revise or modify its proposal and do not constitute discussions.

Colors
See: Scores and Ratings

Contractor
A business enterprise that has entered into a legal arrangement with the government to provide service(s) or product(s). Properly used only after a legal arrangement has been finalized; often used informally to refer to any enterprise that may in the future or has in the past entered into a contract with the government. The words "bidder" or "offeror" are preferred for describing an enterprise with specific intent to participate in the source selection at hand. Prime contractors are those that assume a lead or primary role in the project and are the direct interface with the government. Subcontractors are those that assume a subservient or secondary role in the project and work under the overall

supervision of the prime contractor. Subcontractors generally do not interface directly with the government unless asked to do so by the prime contractor. Sometimes projects are organized so that there is no single prime contractor, but multiple contractors work directly with the government. These contractors are called “associate contractors”. Informally, they are also often called multiple primes. For the purposes of the SDCE, any part of the process that refers to “prime contractor” refers equally to “associate contractor.”

Cost (Price) Criteria
See: Criteria, Cost(Price)

Cost/Financial Risk
See: Risk, Cost/Financial

Criteria, Assessment
A type of evaluation criteria used by evaluators in performing the technical evaluation. Assessment criteria are used in conjunction with evaluation standards to judge how well the offeror’s proposal satisfies the individual specific criteria. Assessment criteria are guidelines that help the source selection evaluators identify strengths, weaknesses, and risks. Typical assessment criteria include “soundness of approach,” “understanding of requirement,” and “compliance with requirement.”

Criteria, Cost (Price)
Evaluation criteria used to determine whether each offeror’s proposed costs are realistic and complete in relation to the solicitation and the technical and management proposals, and to provide an assessment of the reasonableness of the proposed price. These criteria typically include: 1) realism - the compatibility of the proposed costs with proposed scope and effort; 2) completeness - the level of detail the offeror provided in cost data for all RFP requirements and items in the statement of work and the traceability of estimates; 3) reasonableness - the acceptability of the offeror’s methodology used in developing the cost estimates.

Criteria, Evaluation
The basis for measuring each offeror’s ability (as expressed in its proposal) to meet the government’s needs (as they are stated in the solicitation). The evaluation criteria should be tailored to the characteristics of the particular program and should include only those significant aspects expected to have an impact on the ultimate selection decision. Evaluation criteria is an umbrella term that includes cost(price) criteria, specific criteria, and assessment criteria.

Criteria, Model
The basis for evaluating the adequacy of a specific aspect of an offeror’s capability and capacity, against which strengths, weaknesses, and risks are initially identified. The evaluation of capability includes processes, people, tools, and technology.

Criteria, Specific
A type of evaluation criteria that identifies what the customer considers important. They are divided into Areas, as necessary, and into Factors and Subfactors as one moves from lesser to greater detail. The number of levels below each Factor depends upon the complexity of the Factor being evaluated and may vary from Factor to Factor within a given source selection. Common items that might be used for these various levels include supportability, manufacturing, operational utility, design approach, readiness and support, test, project management, reliability and maintainability, system effectiveness, producibility, availability, environmental considerations, technical adequacy, and data management. The term specific criteria is more properly applied to the level of indentures at which the evaluation standards are applied. The term source selection structure is sometimes used to describe the entire hierarchy of Areas, Factors, and Subfactors.

Critical Capability
A set of related model criteria that, when evaluated together, provide the basis for identifying strengths, weaknesses, and risks. CCs represent the lowest level for tailoring of the SDCE model.

Critical Capability Area
A set of related CCs that constitutes an integrated development capability. The CCA facilitates consolidation of strengths, weaknesses, and risks that were identified at the CC level.

Deficiency Report
Deficiency reports document deficiencies found in each offeror's proposal. A "deficiency" is defined as any part of an offeror's proposal which, when compared to the pertinent standard, fails to meet the government's minimum level of compliance. Deficiencies are not derived from a comparative evaluation of the relative strengths and weaknesses of competing offerors' proposals. Resolution of DRs requires that discussions with offerors be opened.

Discussions
Any communication with an offeror which either involves information essential to determine if the offeror's proposal is acceptable or provides the offeror the opportunity to revise or modify its proposal. (See also Deficiency Report and Clarification Request.)

Evaluation Criteria
See: Criteria, Evaluation

Evaluation Standard
Establishes a uniform baseline against which an offeror's solution is compared to determine its value to the government. Evaluation standards can be either quantitative or qualitative. They are written for each of the specific criteria in the source selection structure and are the level at which formal assessments in terms of colors, strengths, weaknesses, and risks are required by the source selection regulations. Items structured below the specific criteria level are evaluated less formally and summarized at the specific criteria level. Items structured above the specific criteria level may or may not be summarized from the lower level(s); but if they are summarized, the strict formality of the specific criteria level must be maintained. The evaluation standards are normally defined and documented prior to the release of the solicitation but no later than the beginning of actual proposal evaluation. The standards are not released to any potential offeror nor to anyone who is not directly involved in the source selection evaluation effort. Evaluation standards can be based entirely on assessments related to the SDCE, can be based entirely on assessments unrelated to the SDCE, or based on a mixture of SDCE and non-SDCE assessments. Attachment 1-2 contains examples of various types of evaluation standards.

Exceptional
See: Scores and Ratings

Factor
See: Specific Criteria

Functional Area
A set of related CCAs functionally grouped into major development capability areas. The collection of CCAs into FAs forms the highest level of the model hierarchy and is used to facilitate the roll-up of information regarding the offeror's strengths, weaknesses, and risks.

General Consideration
An aspect of evaluation in the source selection that typically relates to proposed contractual terms and conditions, results of pre-award surveys, and other surveys or reviews. General considerations combined with use of the evaluation criteria provide an overall integrated assessment that forms the

basis for award. The SDCE is not intended to be used as a general consideration; the results are integrated directly with the other technical evaluation criteria and not used for a pass/fail determination.

Green
See: Scores and Ratings

High
See: Scores and Ratings

Integrated Assessment
Proposal rating, proposal risk, and performance risk are combined (as shown in figure 5-5) with general considerations to provide an integrated assessment that forms the basis for award. The proposal rating, proposal risk, and performance risk assigned to any Factor or Subfactor are independent of each other. Any risk assessment rating may be used with any color rating as appropriate according the evaluation results. Typically, each one has equal weight in the assessment of a given Factor or Subfactor.

Lines of Code
The number of executable computer program statements to be compiled or assembly level instructions to be assembled. Often expressed as KSLOC (thousands of source lines of code). Other measures of program size include function points or numbers of transactions, operations, or operators.

Low
See: Scores and Ratings

Marginal
See: Scores and Ratings

Minor Clarification Request
See: Clarification Request

Model Criteria
See: Criteria, Model

Moderate
See: Scores and Ratings

Narratives
See: Scores and Ratings

Offeror
See: Contractor

Performance Risk
See: Risk, Performance

Performance Risk Assessment Group
A group of experienced government personnel appointed by the SSAC to assess performance risk.

Prime Contractor
See: Contractor

Process Owner
Generally, the office within an organization responsible for a process description and its application. Specifically, the process owner for the SDCE is HQ AFMC/EN.

Procuring Contracting Officer
Contracts may be entered into and signed on behalf of the government only by contracting officers. They also have the authority to administer contracts and make related determinations and findings. They are responsible for ensuring performance of all necessary actions for effective contracting, ensuring compliance with the terms of the contact, and safeguarding the interests of the United States in its contractual relationships.

Program Office
The government organization responsible for the technical, financial, and contractual execution of the program or project under consideration in the source selection. Primary duties, relative to source selection, are the development of the acquisition plan, preparation of the source selection plan (including the source selection structure and evaluation criteria), and development of the evaluation standards.

Project Office
See: Program Office

Proposal Rating
See: Scores and Ratings

Proposal Risk
See: Risk, Proposal

Qualitative Standard
See: Evaluation Standard

Quantitative Standard
See: Evaluation Standard

Ratings and Scores
See: Scores and Ratings

Red
See: Scores and Ratings

Risk, Cost/Financial
Those risks associated with economic and cost impacts of the proposed approach. One aspect of this risk determination is the proposed costs compared with the program office assessment or the independent cost assessment if one was performed.

Risk, Performance
The assessment of an offeror's present and past work record to assess confidence in the offeror's ability to successfully perform as proposed. The key measure is how much doubt exists, based on the offeror's performance record, that the offeror can perform the proposed effort. Performance risk is discussed in the evaluation narratives along with the strengths and weaknesses and is depicted in briefings with the color ratings.

Risk, Proposal
The identification and assessment of the risks associated with an offeror's proposed approach as it relates to accomplishing the requirements of the solicitation. Proposal risk includes technical risk, schedule risk, and cost/financial risk. Risks may be inherent in a proposed approach by virtue of its relationship to the state of the art. Risks may occur as a result of a particular technical approach, manufacturing plan, selection of materials, processes, equipment, etc., or as a result of cost, schedule, and economic impacts associated with these approaches. Risk may also occur from the impact that these will have on the offeror's ability to perform in view of its technical approach. The prime's proposed subcontract arrangements may also impact proposal risk. (See also Scores and Ratings.)

Risk, Schedule
Those risks associated with the schedule as evidenced in the interrelationship of deliveries and milestones and the amount and distribution of slack and reserve. Also based on the estimated ability of the offeror to meet the incremental and final deliveries and milestones.

Risk, Technical
Those risks associated with the technical aspects of the program being proposed, including those related to technical approach relative to the state of the art, manufacturing, materials, processes, and equipment.

Schedule Risk
See: Risk, Schedule

Scores and Ratings
Any of several classes of indicators used in the evaluation of the various aspects of the proposals or used in the narrative write-ups that become part of the evaluation report. The rating system is structured to evaluate each offeror's proposal relative to the requirements as well as the strengths, weaknesses, and risks associated with that proposal. The objective of the rating system is to display an assessment of all important aspects of the offerors' proposals. The narrative is the principal means available to the SSAC to perform a comparative analysis. Clarity and brevity are the keys to successfully prepared narratives. The narrative must indicate, as a minimum, what is offered, whether it meets or fails to meet the standard, any strengths or weaknesses, the evaluator's opinions of what may be done to remedy deficiencies, impacts of any deficiencies, and a risk assessment of the offeror's proposal approach and ability to perform. The vocabulary and definitions of the rating system can be arranged in three categories as explained in table B-1.

SDCE Team
The personnel actually performing the SDCE analysis and site visits. Preferably all, and at least some, of the members of this team are also members of the SSEB.

Significant Clarification Request
See: Clarification Request

Site Visit
A visit by the acquisition source selection organization to a bidder's facility during the source selection process to confirm an understanding of proposed capability and capacity.

Source Selection
The formal process by which the government evaluates the competing proposals in an impartial, equitable, and comprehensive manner. The objective is to select the source (offeror) whose proposal has the highest degree of credibility and whose performance can be expected to best meet the government's requirements at an affordable cost. Air Force source selection awards are based on an

Table B-1. Scores and Ratings

Rating	Description
	Value Judgments
Strength	A significant, outstanding, or exceptional aspect of an offeror's proposal that exceeds the evaluation standard and provides a useful capability that will be included in the specification or statement of work, or is inherent in the offeror's process, so that the government will be assured of receiving the benefit under the resultant contract.
Weakness	An aspect of or omission from an offeror's proposal that contributes to a deficiency in meeting an evaluation standard or is otherwise a shortcoming of the proposal that has the potential to degrade contract performance.
Acceptable	Simply satisfactory, neither a strength nor a weakness.
	Proposal Ratings
Blue/Exceptional	Exceeds specified performance or capability in a beneficial way to the government, and has no significant weakness.
Green/Acceptable	Meets evaluation standards, and any weaknesses are readily corrected.
Yellow/Marginal	Fails to meet evaluation standards; however, any significant deficiencies are correctable.
Red/Unacceptable	Fails to meet a minimum requirement of the RFP, and the deficiency is uncorrectable without a major revision of the proposal.
	Proposal Risk Assessments
Low	Has little potential to cause disruption of schedule, increase in cost, or degradation of performance. Normal contractor effort and normal government monitoring will probably be able to overcome difficulties.
Moderate	Can potentially cause some disruption of schedule, increase in cost, or degradation of performance. However, special contractor emphasis and close government monitoring will probably be able to overcome difficulties.
High	Likely to cause significant serious disruption of schedule, increase in cost, or degradation of performance, even with special contractor emphasis and close government monitoring.

integrated assessment of each offeror's cost (price) criteria, specific criteria, assessment criteria, proposal risk, performance risk, and general considerations. This process is documented in AFFARS AA.

Source Selection Advisory Council
A group of senior government personnel appointed by the SSA to provide counsel during the source selection process and to prepare for the SSA a comparative analysis of the evaluation results of the SSEB. The SSAC is staffed with personnel possessing broad experience in fields such as system development, systems engineering, manufacturing, operational requirements, finance, logistics, law, and contracting.

Source Selection Authority
The official designated to direct the source selection process and make the source selection decision.

Source Selection Evaluation Board
A group of government personnel representing the various functional and technical disciplines relevant to the acquisition that evaluates proposals and reports its findings to the SSAC. May also include non-government personnel under contract to the government who may be called on to furnish expert advice. The primary responsibilities of the SSEB include the conduct of an in-depth review and evaluation of each proposal against the solicitation requirements, the approved evaluation criteria, and the standards. The SSEB submits its evaluation to the SSAC. Only fully qualified personnel possessing the professional skills and knowledge required for an objective evaluation and assessment of offerors' proposals are selected to participate on the SSEB. The Program Manager is usually designated the SSEB chairperson.

Source Selection Evaluation Guide
An optional document developed by and for an SSEB that is used to conveniently document the standards for proposal evaluation and detailed procedures and administrative guidance for conducting the evaluation. The SSEG is unique for each acquisition.

Source Selection Evaluation Team
An alternative source selection organization combining the functions of the SSAC and SSEB.

Source Selection Plan
A plan, approved by the SSA, that describes how the source selection team is organized, how the proposals will be evaluated and analyzed, and how the source or sources will be selected.

Source Selection Structure
See: Criteria, Specific

Specific Criteria
See: Criteria, Specific

Strength
See: Scores and Ratings

Subcontractor
See: Contractor

Subfactor
See: Specific Criteria

Technical Risk
See: Risk, Technical

Unacceptable
See: Scores and Ratings

Weakness
See: Scores and Ratings

Yellow
See: Scores and Ratings

Section C. References

Nomenclature	Title/Description
DoDD 5000.1	**Defense Acquisitions** Establishes a disciplined management approach for acquiring systems and materiel that satisfy the operational user's needs.
DoDI 5000.2	**Defense Acquisition Management Policies and Procedures** Establishes an integrated framework for translating broadly stated mission needs into stable, affordable acquisition programs that meet the operational user's needs and can be sustained, given projected resource constraints.
DoDM 4245.7	**Transition from Development to Production** Provides assistance in structuring technically sound programs, assessing their risk, and identifying areas needing corrective action. Designed to facilitate the discipline needed to make wise decisions. Provides numerous templates that identify and describe areas of risk and suggested technical methods for reducing that risk.
DoD-STD-1467	**Software Support Environment** Defines the efforts necessary to ensure the existence of a com-plete life-cycle software support capability for the contractually deliverable software when it enters the operational inventory.
DoD-STD-2167A	**Defense System Software Development** Establishes requirements to be applied during the acquisition, development, or support of software systems.
DoD-STD-2168	**Defense System Software Quality Program** Establishes requirements for a software quality program to be applied during the acquisition, development, or support of software systems.
MIL-STD-480B	**Configuration Control – Engineering Changes, Deviations and Waivers** Provides requirements for maintaining configuration control and preparing engineering changes, waivers, and revisions.
MIL-STD-483A	**Configuration Management Practices for Systems, Equipment, Munitions, and Computer Programs** Prescribes requirements of configuration management practices.
MIL-STD-490A	**Specification Practices** Sets forth uniform practices for the preparation, interpretation, change, and revision of program-peculiar specifications.

Nomenclature	Title/Description
MIL-STD-490B	**(Draft) Preparation of Program-Unique Specifications** Establishes formats, contents, and procedures for the preparation of specifications for program-unique configuration items, processes, and materials. Purpose is to establish uniform practices for specification preparation, to ensure the inclusion of essential requirements, and to aid in the use and analysis of specification content.
MIL-STD-498	**(Draft) Software Development and Documentation** Merges previous/existing standards; intended to replace DoD-STD-2167A, DoD-STD-7935A, and DoD-STD-1703. Eliminates the distinction between weapon system and automated information system software development requirements. Key changes are the introduction of a harmonized document set; improved compatibility with incremental and evolutionary development models; improved compatibility with non-hierarchical design methods, such as object oriented; improved compatibility with computer-aided software engineering tools; alternatives to, and more flexibility in, preparing documents; clearer requirements for incorporating reusable software; introduction of software management indicators; added emphasis on software supportability; and improved links to systems engineering.
MIL-STD-499B	**(Draft) Systems Engineering** Intent is to assist in defining, performing, managing, and evaluating systems engineering efforts in defense system acquisitions and technology developments. Implements the technical essence of Concurrent Engineering/ Integrated Product and Process Development. Defines the purpose and use of the SEMP and SEMS.
MIL-STD-881A	**Work Breakdown Structures for Defense Material Items** Establishes uniform criteria for preparing and using work breakdown structures during the acquisition of defense systems and materiel items.
MIL-STD-973	**Configuration Management** Draft standard consolidating configuration management requirements. When approved, will replace MIL-STD-480, -483, and -1521.
MIL-STD-1521B	**Technical Reviews and Audits for Systems, Equipment, and Computer Software** Establishes the requirements for the technical reviews and audits for defense acquisitions.

Nomenclature	Title/Description
MIL-STD-1803	**Software Development Integrity Program** Provides general requirements and specific tasks to achieve software integrity during the development and deployment of systems and equipment. This standard, when used in conjunction with DoD-STD-2167A, forms the basis for the Software Development Integrity Program and is intended to improve the performance and supportability of military software. It puts integrity on a par with cost, schedule, and other design and performance criteria throughout the acquisition cycle.
MIL-STD-1815A	**Ada® Programming Language** Specifies the form and meaning of program units written in Ada.
MIL-HDBK-347	**MCCR Software Support** Presents software support concepts, procedures, and guidance to all managers responsible for mission-critical computer resource development or support.
MIL-HDBK-499-3	**(Draft) Systems Engineering/Configuration Management Life Cycle Application** Based on MIL-STD-499B; describes a model for the application of systems engineering throughout the system life cycle. The model was developed by AFMC in support of Integrated Weapon Systems activities.
MIL-HDBK-782	**Software Support Environment Acquisition** Provides implementation guidance for DoD-STD-1467.
AFAC 92-33	**Air Force FAR Supplement (AFFARS) Appendix AA, Formal Source Selection for Major Acquisitions** Describes the terms, objectives, philosophy, and procedures to be followed for major Air Force source selections. This is the most critical reference for this pamphlet. To ensure the precise use of terms necessary for the adequate integration of the SDCE into source selection, many quotes and paraphrases from this reference are used throughout this pamphlet.
AFR 800-2	**Acquisition Program Management** Establishes the policy for managing all Air Force acquisition and modification programs funded either through procurement appropriations, the Security Assistance Program, or the Research, Development, Test, and Evaluation appropriation.
AFR 800-14	**Lifecycle Management of Computer Resources in Systems** Establishes policy for the acquisition and support of computer resources acquired under the program management concept of AFR 800-2.

<u>Nomenclature</u>	<u>Title/Description</u>
AFMCPAM 63-101	**Acquisition Risk Management Guide** Lays out a generic risk management process that program offices can adapt to their own circumstances.
AFMCP 800-51	**Software Development Capability Assessment (SDCA)** Describes methods to assess and evaluate a prospective contractor's software development capability.
AFMCP 800-60	**Integrated Weapon Systems Management (IWSM) Guide** Describes the implementation of the IWSM management philosophy for all programs managed by Headquarters Air Force Materiel Command. Modifies many processes and establishes some new ones. These changes will eventually percolate through most of the other documents listed in this section.
ASDP 800-5	**Software Development Capability/Capacity Review** Provides guidance for planning and conducting the Software Development Capability/Capacity Review as an integral part of the system/subsystem acquisition source selection process.
ISO 9001:1987(E)	**International Standard, Quality Systems Model for Quality Assurance in Design/Development, Production, Installation, and Servicing** Specifies quality system requirements for use where a contract between two parties requires the demonstration of a supplier's capability to design and supply product. The requirements specified in this international standard are aimed primarily at preventing nonconformity at all stages from design through servicing.
ISO 9001-3:1991(E)	**International Standard, Quality Systems – Part 3: Guidelines for the Application of ISO 9001 to the Development, Supply, and Maintenance of Software** Companion volume to ISO 9001. Sets out guidelines to facilitate the application to organizations developing, supplying, and maintaining software. The guidelines describe the suggested controls and methods for producing software which meets purchaser's requirements.
CMU/SEI-93-TR-24	**Software Engineering Institute, Capability Maturity Model (CMM) Software, Version 1.1** Provides a technical overview of the CMM for software and describes the process maturity framework, the structure of the CMM, and how the CMM is used in practice. Provides an introduction to how the CMM is related to software process assessments, software capability evaluations, and process improvement programs.

Nomenclature	Title/Description
CMU/SEI-93-TR-25	**Software Engineering Institute, Key Practices of the Capability Maturity Model, Version 1.1** Companion document to CMU/SEI-93-TR-24. Describes, in detail, the key practices that correspond to the key process areas at each maturity level of the CMM and information on how to interpret the key practices.
5169/T1–Issue 1.0	**Cranfield IT Institute and UK Ministry of Defence, ImproveIT** Initiative directed toward continuous process and quality improvement matched to business needs. Focused around an international standard on process management which will itself provide a framework for a capability assessment scheme with a dual mode of operation: a) a process assessment mode to facilitate general levels of process improvement in industry as a whole, and b) a capability evaluation mode for use in procurement in evaluating contract risk. Includes concepts, requirements, and definitions. Describes and compares numerous existing or emerging methods.
TRILLIUM	**(Draft 2.2) Bell Canada, Telecom Software Product Development Capability Assessment Model** Adapts and extends the SEI CMM for application to embedded software systems (such as telecommunication systems) with a heavy system perspective. Major enhancements include: a) introduction of a stronger customer focus; b) broadening of scope to encompass the entire product [hardware, software, system, documentation, training, and services]; c) inclusion of ISO, IEEE, Bellcore, and IEC standards; d) addition of technological maturity; e) a focus on common-sense and practice over theory; and f) description of a roadmap approach to improvements.
SDCE Database	**SDCE PAT Contractor/Government User Data Gathering** A collection of notes and reports detailing observations, issues, and concerns with the current state of the practice. Gathered from industry and government experts in 1992 and 1993.

JAMES F. BAIR
Director of Engineering and
Technical Management